z/OS Communications Server
Version 2 Release 3

IP Messages:
Volume 2 (EZB, EZD)

IBM
Contents

Figures.................................................................................................................................................. v

About this document................................................................................................................................vii

Summary of changes for IP Messages: Volume 2 (EZB, EZD).............................................................. xv

Chapter 1. IP message standards introduction...................................................................................... 1

Chapter 2. EZB0xxxx messages........................................................................................................... 7

Chapter 3. EZB1xxxx messages.......................................................................................................... 145

Chapter 4. EZB2xxxx messages........................................................................................................ 177

Chapter 5. EZB3xxxx messages.......................................................................................................... 277

Chapter 6. EZB4xxxx messages........................................................................................................ 423

Chapter 7. EZB6xxxx messages.......................................................................................................... 483

Chapter 8. EZB8xxxx messages.......................................................................................................... 485

Chapter 9. EZB9xxxx messages.......................................................................................................... 487

Chapter 10. EZBHxxxx messages....................................................................................................... 489

Chapter 11. EZD0xxxx messages........................................................................................................ 513

Chapter 12. EZD1xxxx messages........................................................................................................ 683

Chapter 13. EZD2xxxx messages........................................................................................................ 1323

Appendix A. Related protocol specifications...................................................................................... 1365

Appendix B. Accessibility.................................................................................................................... 1385

Notices.................................................................................................................................................. 1387

Bibliography......................................................................................................................................... 1391

Communicating your comments to IBM............................................................................................. 1397
Figures

1. Sample IP message format........................................................................................................................... 1
2. Sample IP message identifier.......................................................................................................................... 1
About this document

This document describes the Internet Protocol (IP) messages that occur in z/OS Communications Server. The information in this document supports both IPv6 and IPv4. Unless explicitly noted, information describes IPv4 networking protocol. IPv6 support is qualified in the text.

For information about how to set up, initialize, and customize your Transmission Control Protocol/Internet Protocol (TCP/IP) services system, see the z/OS Communications Server: IP Configuration Reference, the z/OS Communications Server: IP Configuration Guide and the z/OS Communications Server: IP Programmers Guide and Reference. For information about how to use the applications on your TCP/IP system, see z/OS Communications Server: IP User’s Guide and Commands.

This document refers to Communications Server data sets by their default SMP/E distribution library name. Your installation might, however, have different names for these data sets where allowed by SMP/E, your installation personnel, or administration staff. For instance, this document refers to samples in SEZAINST library as simply in SEZAINST. Your installation might choose a data set name of SYS1.SEZAINST, CS390.SEZAINST or other high level qualifiers for the data set name.

Who should read this document

This document assists TCP/IP operators, system programmers, and users to:

- Analyze a problem
- Classify the problem as a specific type
- Describe the problem to the IBM® Software Support Center

Familiarity with TCP/IP concepts and terms is assumed.

How this document is organized

The messages are listed in alphanumeric order by message ID. For each message ID, the books contains the text and a description of the message. This book contains the following chapters:

- Chapter 2, “EZB0xxxx messages,” on page 7 contains messages in the EZB0xxxx range.
- Chapter 3, “EZB1xxxx messages,” on page 145 contains messages in the EZB1xxxx range.
- Chapter 4, “EZB2xxxx messages,” on page 177 contains messages in the EZB2xxxx range.
- Chapter 5, “EZB3xxxx messages,” on page 277 contains messages in the EZB3xxxx range.
- Chapter 6, “EZB4xxxx messages,” on page 423 contains messages in the EZB4xxxx range.
- Chapter 7, “EZB6xxxx messages,” on page 483 contains messages in the EZB6xxxx range.
- Chapter 9, “EZB9xxxx messages,” on page 487 contains common messages that are called by several application and function components. These messages are in the EZB9xxxx range.
- Chapter 10, “EZBHxxxx messages,” on page 489 contains IBM Health Checker for z/OS messages in the EZBHxxxx range.
- Chapter 11, “EZD0xxxx messages,” on page 513 contains messages in the EZD0xxxx range.
- Chapter 12, “EZD1xxxx messages,” on page 683 contains messages in the EZD1xxxx range.
- Chapter 13, “EZD2xxxx messages,” on page 1323 contains messages in the EZD2xxxx range.
- Appendix A, “Related protocol specifications,” on page 1365 lists the related protocol specifications for TCP/IP.
How to use this document

To use this document, you should be familiar with z/OS TCP/IP Services and the TCP/IP suite of protocols.

How to contact IBM service

For immediate assistance, visit this website: http://www.software.ibm.com/support

Most problems can be resolved at this website, where you can submit questions and problem reports electronically, and access a variety of diagnosis information.

For telephone assistance in problem diagnosis and resolution (in the United States or Puerto Rico), call the IBM Software Support Center anytime (1-800-IBM-SERV). You will receive a return call within 8 business hours (Monday – Friday, 8:00 a.m. – 5:00 p.m., local customer time).

Outside the United States or Puerto Rico, contact your local IBM representative or your authorized IBM supplier.

If you would like to provide feedback on this publication, see “Communicating your comments to IBM” on page 1397.

Conventions and terminology that are used in this information

Commands in this information that can be used in both TSO and z/OS UNIX environments use the following conventions:

• When describing how to use the command in a TSO environment, the command is presented in uppercase (for example, NETSTAT).

• When describing how to use the command in a z/OS UNIX environment, the command is presented in bold lowercase (for example, netstat).

• When referring to the command in a general way in text, the command is presented with an initial capital letter (for example, Netstat).

All the exit routines described in this information are installation-wide exit routines. The installation-wide exit routines also called installation-wide exits, exit routines, and exits throughout this information.

The TPF logon manager, although included with VTAM®, is an application program; therefore, the logon manager is documented separately from VTAM.

Samples used in this information might not be updated for each release. Evaluate a sample carefully before applying it to your system.

Note: In this information, you might see the following Shared Memory Communications over Remote Direct Memory Access (SMC-R) terminology:

• RoCE Express®, which is a generic term representing IBM 10 GbE RoCE Express, IBM 10 GbE RoCE Express2, and IBM 25 GbE RoCE Express2 feature capabilities. When this term is used in this information, the processing being described applies to both features. If processing is applicable to only one feature, the full terminology, for instance, IBM 10 GbE RoCE Express will be used.

• RoCE Express2, which is a generic term representing an IBM RoCE Express2® feature that might operate in either 10 GbE or 25 GbE link speed. When this term is used in this information, the processing being described applies to either link speed. If processing is applicable to only one link speed, the full terminology, for instance, IBM 25 GbE RoCE Express2 will be used.

• RDMA network interface card (RNIC), which is used to refer to the IBM® 10 GbE RoCE Express, IBM® 10 GbE RoCE Express2, or IBM 25 GbE RoCE Express2 feature.
• Shared RoCE environment, which means that the "RoCE Express" feature can be used concurrently, or shared, by multiple operating system instances. The feature is considered to operate in a shared RoCE environment even if you use it with a single operating system instance.

Clarification of notes
Information traditionally qualified as Notes is further qualified as follows:

Attention
Indicate the possibility of damage

Guideline
Customary way to perform a procedure

Note
Supplemental detail

Rule
Something you must do; limitations on your actions

Restriction
Indicates certain conditions are not supported; limitations on a product or facility

Requirement
Dependencies, prerequisites

Result
Indicates the outcome

Tip
Offers shortcuts or alternative ways of performing an action; a hint

Prerequisite and related information

z/OS Communications Server function is described in the z/OS Communications Server library. Descriptions of those documents are listed in “Bibliography” on page 1391, in the back of this document.

Required information
Before using this product, you should be familiar with TCP/IP, VTAM, MVS™, and UNIX System Services.

Softcopy information
Softcopy publications are available in the following collection.

<table>
<thead>
<tr>
<th>Titles</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Z Redbooks</td>
<td>The IBM Z® subject areas range from e-business application development and enablement to hardware, networking, Linux, solutions, security, parallel sysplex, and many others. For more information about the Redbooks® publications, see <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> and <a href="http://www.ibm.com/systems/z/os/zos/zfavorites/">http://www.ibm.com/systems/z/os/zos/zfavorites/</a>.</td>
</tr>
</tbody>
</table>

Other documents
This information explains how z/OS references information in other documents.
When possible, this information uses cross-document links that go directly to the topic in reference using shortened versions of the document title. For complete titles and order numbers of the documents for all products that are part of z/OS, see z/OS Information Roadmap (SA23-2299). The Roadmap describes what level of documents are supplied with each release of z/OS Communications Server, and also describes each z/OS publication.

To find the complete z/OS library, visit the z/OS library in IBM Knowledge Center (www.ibm.com/support/knowledgecenter/SSLTBW/welcome).

Relevant RFCs are listed in an appendix of the IP documents. Architectural specifications for the SNA protocol are listed in an appendix of the SNA documents.

The following table lists documents that might be helpful to readers.

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNA Formats</td>
<td>GA27-3136</td>
</tr>
<tr>
<td>TCP/IP Tutorial and Technical Overview</td>
<td>GG24-3376</td>
</tr>
<tr>
<td>Understanding LDAP</td>
<td>SG24-4986</td>
</tr>
<tr>
<td>z/OS Cryptographic Services System SSL Programming</td>
<td>SC14-7495</td>
</tr>
<tr>
<td>z/OS IBM Tivoli Directory Server Administration and Use for z/OS</td>
<td>SC23-6788</td>
</tr>
<tr>
<td>z/OS JES2 Initialization and Tuning Guide</td>
<td>SA32-0991</td>
</tr>
<tr>
<td>z/OS Problem Management</td>
<td>SC23-6844</td>
</tr>
<tr>
<td>z/OS MVS Diagnosis: Reference</td>
<td>GA32-0904</td>
</tr>
<tr>
<td>z/OS MVS Diagnosis: Tools and Service Aids</td>
<td>GA32-0905</td>
</tr>
<tr>
<td>z/OS MVS Using the Subsystem Interface</td>
<td>SA38-0679</td>
</tr>
<tr>
<td>z/OS Program Directory</td>
<td>GI11-9848</td>
</tr>
<tr>
<td>z/OS UNIX System Services Command Reference</td>
<td>SA23-2280</td>
</tr>
<tr>
<td>z/OS UNIX System Services Planning</td>
<td>GA32-0884</td>
</tr>
<tr>
<td>z/OS UNIX System Services Programming: Assembler Callable Services Reference</td>
<td>SA23-2281</td>
</tr>
<tr>
<td>z/OS UNIX System Services User's Guide</td>
<td>SA23-2279</td>
</tr>
<tr>
<td>z/OS XL C/C++ Runtime Library Reference</td>
<td>SC14-7314</td>
</tr>
</tbody>
</table>
Redbooks publications

The following Redbooks publications might help you as you implement z/OS Communications Server.

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM z/OS Communications Server TCP/IP Implementation, Volume 1: Base Functions, Connectivity, and Routing</td>
<td>SG24-8096</td>
</tr>
<tr>
<td>IBM z/OS Communications Server TCP/IP Implementation, Volume 2: Standard Applications</td>
<td>SG24-8097</td>
</tr>
<tr>
<td>IBM z/OS Communications Server TCP/IP Implementation, Volume 3: High Availability, Scalability, and Performance</td>
<td>SG24-8098</td>
</tr>
<tr>
<td>IBM z/OS Communications Server TCP/IP Implementation, Volume 4: Security and Policy-Based Networking</td>
<td>SG24-8099</td>
</tr>
<tr>
<td>IBM Communication Controller Migration Guide</td>
<td>SG24-6298</td>
</tr>
<tr>
<td>IP Network Design Guide</td>
<td>SG24-2580</td>
</tr>
<tr>
<td>Managing OS/390 TCP/IP with SNMP</td>
<td>SG24-5866</td>
</tr>
<tr>
<td>Migrating Subarea Networks to an IP Infrastructure Using Enterprise Extender</td>
<td>SG24-5957</td>
</tr>
<tr>
<td>SecureWay Communications Server for OS/390 V2R8 TCP/IP: Guide to Enhancements</td>
<td>SG24-5631</td>
</tr>
<tr>
<td>SNA and TCP/IP Integration</td>
<td>SG24-5291</td>
</tr>
<tr>
<td>TCP/IP in a Sysplex</td>
<td>SG24-5235</td>
</tr>
<tr>
<td>TCP/IP Tutorial and Technical Overview</td>
<td>GG24-3376</td>
</tr>
<tr>
<td>Threadsafe Considerations for CICS</td>
<td>SG24-6351</td>
</tr>
</tbody>
</table>

Where to find related information on the Internet

z/OS

This site provides information about z/OS Communications Server release availability, migration information, downloads, and links to information about z/OS technology

http://www.ibm.com/systems/z/os/zos/

z/OS Internet Library

Use this site to view and download z/OS Communications Server documentation


IBM Communications Server product

The primary home page for information about z/OS Communications Server


z/OS Communications Server product

The page contains z/OS Communications Server product introduction

IBM Communications Server product support
Use this site to submit and track problems and search the z/OS Communications Server knowledge base for Technotes, FAQs, white papers, and other z/OS Communications Server information
http://www.software.ibm.com/support

IBM Communications Server performance information
This site contains links to the most recent Communications Server performance reports
http://www.ibm.com/support/docview.wss?uid=swg27005524

IBM Systems Center publications
Use this site to view and order Redbooks publications, Redpapers, and Technotes
http://www.redbooks.ibm.com/

IBM Systems Center flashes
Search the Technical Sales Library for Techdocs (including Flashes, presentations, Technotes, FAQs, white papers, Customer Support Plans, and Skills Transfer information)
http://www.ibm.com/support/techdocs/atsmastr.nsf

Tivoli® NetView® for z/OS
Use this site to view and download product documentation about Tivoli NetView for z/OS
http://www.ibm.com/support/knowledgecenter/SSZJDU/welcome

RFCs
Search for and view Request for Comments documents in this section of the Internet Engineering Task Force website, with links to the RFC repository and the IETF Working Groups web page
http://www.ietf.org/rfc.html

Internet drafts
View Internet-Drafts, which are working documents of the Internet Engineering Task Force (IETF) and other groups, in this section of the Internet Engineering Task Force website
http://www.ietf.org/ID.html

Information about web addresses can also be found in information APAR II11334.

Note: Any pointers in this publication to websites are provided for convenience only and do not serve as an endorsement of these websites.

DNS websites
For more information about DNS, see the following USENET news groups and mailing addresses:

USENET news groups
comp.protocols.dns.bind

BIND mailing lists
https://lists.isc.org/mailman/listinfo

BIND Users
- Subscribe by sending mail to bind-users-request@isc.org.
- Submit questions or answers to this forum by sending mail to bind-users@isc.org.

BIND 9 Users (This list might not be maintained indefinitely.)
- Subscribe by sending mail to bind9-users-request@isc.org.
- Submit questions or answers to this forum by sending mail to bind9-users@isc.org.
The z/OS Basic Skills Information Center

The z/OS Basic Skills Information Center is a web-based information resource intended to help users learn the basic concepts of z/OS, the operating system that runs most of the IBM mainframe computers in use today. The Information Center is designed to introduce a new generation of Information Technology professionals to basic concepts and help them prepare for a career as a z/OS professional, such as a z/OS systems programmer.

Specifically, the z/OS Basic Skills Information Center is intended to achieve the following objectives:

- Provide basic education and information about z/OS without charge
- Shorten the time it takes for people to become productive on the mainframe
- Make it easier for new people to learn z/OS

To access the z/OS Basic Skills Information Center, open your web browser to the following website, which is available to all users (no login required): https://www.ibm.com/support/knowledgecenter/zosbasics/com.ibm.zos.zbasics/homepage.html?cp=zosbasics
Summary of changes for IP Messages: Volume 2 (EZB, EZD)

This document contains terminology, maintenance, and editorial changes, including changes to improve consistency and retrievability. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

Changes made in z/OS Communications Server Version 2 Release 3

This document contains information previously presented in z/OS Communications Server: IP Messages Volume 2 (EZB, EZD), which supported z/OS Version 2 Release 2.

December 2018
Changed information
• EZD2044I

March 2018
Changed information
• EZD2028I

January 2018
New information
• EZD2043I

September 2017
New information
• EZD0840I
• EZD1975E
• EZD2042I

Changed information
• EZD2026I

Changes made in z/OS Version 2 Release 2

This document contains information previously presented in z/OS Communications Server: IP Messages Volume 2 (EZB, EZD), SC27-3655-01, which supported z/OS Version 2 Release 1.

New information
• EZD1591I
• EZD2030I
• EZD2031I
• EZD2032I
• EZD2033I
• EZD2034I
• EZD2040I
• EZD2041I

**Changed information**
• EZB2460E
• EZBH008E

**Deleted information**
• EZB8801I

**Changes made in z/OS Version 2 Release 1, as updated December 2013**
This document contains information previously presented in z/OS Communications Server: IP Messages Volume 2 (EZB, EZD), SC27-3655-00, which supported z/OS Version 2 Release 1.

**New information**
• EZD2029I

**z/OS Version 2 Release 1 summary of changes**
See the Version 2 Release 1 (V2R1) versions of the following publications for all enhancements related to z/OS V2R1:
• z/OS Migration
• z/OS Planning for Installation
• z/OS Summary of Message and Interface Changes
• z/OS Introduction and Release Guide
Chapter 1. IP message standards introduction

This topic contains the following information about IP message standards:

- “Message text formats” on page 1
- “Message description formats” on page 3
- “Message routing codes” on page 3
- “Message descriptor codes” on page 4
- “Message groups” on page 5

Message text formats

Most IP messages are preceded by an identifier, as illustrated in Figure 1 on page 1.

![Figure 1. Sample IP message format](image)

Message identifiers

All message identifiers include the following sections:

- Prefix
- Message number
- Message type code

See Figure 2 on page 1 for a sample IP message identifier.

![Figure 2. Sample IP message identifier](image)

Prefix

Message identifiers include a prefix that identifies the source of the message. The following message prefixes are used by TCP/IP and its associated applications:

- EZA
Message number
Message identifiers include a unique 2- through 4-digit message number.

Message type code
The following type codes are used in IP messages:

**A Action**
The message indicates that an action is required.

**E Eventual Action**
You must eventually take some action to correct a problem. The system continues processing without waiting for a response.

**I Information**
The message is for your information. This type code can be used to notify you of an error. No response is necessary, but you might need to take some action.

**S Severe Error**
The message is for a system programmer.

**W Wait**
Processing stops until the operator takes a required action.

Syntax notation in message text
In this documentation, IP messages are described with the following syntax notation:

**Non-highlighted characters**
Represent the actual text of the message.

**Italic characters**
Represent message variables. The variables are replaced by their values in the actual message.

**Braces { }**
Represent a group of text strings, only one of which is displayed in the actual message. The text strings are separated by or-signs (|) in the braces.
The braces and or-signs are not displayed in the actual message.

**Brackets [ ]**
Represent optional messages or optional parts of a message. Optional messages or optional parts of a message are displayed only under certain circumstances that are described in the "Explanation" section of the message. If an optional part has more than one possible value, or-signs separate the possibilities.
The brackets and or-signs are not displayed in the actual message.
Message description formats

A message consists of several sections. Not all sections are used for each message. For messages that are issued as a group, the "Explanation" section of the first message usually contains a complete description of the other messages in the group.

**Explanation**
- Explains why the message was issued and describes all text and variables in the message.

**System action**
- Explains the system state after the message was issued. This section also indicates whether the system is waiting for a reply.

**Operator response**
- Describes actions that the operator can or must take at the console.

**System programmer response**
- Suggests actions, programming changes, or system definition changes that isolate or correct errors or improve the efficiency of the system.

**User response**
- Describes actions that the user can or must take at the terminal.

**Problem determination**
- Additional instructions for determining the cause of the problem, searching problem databases, and if necessary, reporting the problem to the IBM support center. These instructions are for system programmers who can troubleshoot problems.

**Source**
- Element, product, or component that issued the message.

**Module**
- Module or modules that issued the message.

**Automation**
- Indicates whether the message is a candidate for automation.

**Example**
- Example of the message with variable fields replaced with actual values, perhaps in context with other messages.

Message routing codes

Routing codes determine where a message is displayed. More than one routing code might be assigned to the message. With multiple-console support, each console operator receives the messages related only to the commands entered at that console or to the functions assigned to that console, regardless of the routing codes assigned to those messages. If a message that is routed to a particular console cannot be issued at that console, that message is issued at the master console.

The following routing codes are used in IP messages:

**Code**

<table>
<thead>
<tr>
<th>Code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Master Console Action</strong>: This message indicates a change in the system status and demands action by the master console operator.</td>
</tr>
<tr>
<td>2</td>
<td><strong>Master Console Information</strong>: This message indicates a change in the system status. Such a message does not demand action, but alerts the master console operator to a condition that might require action. This routing code is used for any message that indicates job status, and also for processor and problem program messages to the master console operator.</td>
</tr>
</tbody>
</table>
Tape Pool: This message specifies the status of a tape unit or reel, the disposition of a tape reel, or other tape-oriented information. For example, this can be a message which requests that tapes be mounted.

Direct Access Pool: This message specifies the status of a direct access unit or pack, the disposition of a disk pack, or other direct-access-oriented information. For example, this can be a message which requests that disks be mounted.

Tape Library: This message specifies the tape library information. For example, this can be a message which requests, by volume serial numbers, that tapes be obtained for system or programmer use.

Disk Library: This message specifies the disk library information. For example, this can be a message which requests, by volume serial numbers, that disk packs be obtained for system or programmer use.

Unit Record Pool: This message specifies the unit-record equipment information. For example, this can be a message which requests that printer trains be mounted.

Teleprocessing Control: This message specifies the status or the disposition of data communication equipment. For example, this can be a message that indicates line errors.

System Security: This message is associated with security checking. For example, this can be a message that requires a reply that is specifying a password.

System Error Maintenance: This message indicates either a system error, or an input/output error that cannot be corrected. It also indicates a message that is associated with system maintenance.

Programmer Information: This message is for the problem programmer. This routing code is used only when the program that issued the message cannot route the message to the programmer by using the system-output data set facility. The message is displayed in the system output message class of the job.

Emulators: This message is issued by an emulator program.

Reserved for customer use.

Reserved for customer use.

Reserved for customer use.

Reserved for future expansion.

Message descriptor codes

Descriptor codes describe the kind of message being issued. These codes, with message routing codes, determine how a message is to be printed or displayed and how a message is to be deleted from a display device. Descriptor codes 1 – 7 are mutually exclusive; only one such code is assigned to a message. Descriptor codes 8 – 10 can be displayed with any other descriptor codes.

The following descriptor codes are used in IP messages:

<table>
<thead>
<tr>
<th>Code</th>
<th>Meaning</th>
</tr>
</thead>
</table>

1 **System Failure**: This message indicates that an error that cannot be corrected occurs. To continue, the operator must restart the system.

2 **Immediate Action Required**: This message requires an immediate action by the operator. The action is required because the message issuer is in a wait state until the action is taken, or because system performance is degraded until the action is taken.

3 **Eventual Action Required**: This message requires an eventual action by the operator. The task does not await completion of the action.

4 **System Status**: This message indicates the status of a system task or of a hardware unit.

5 **Immediate Command Response**: This message is issued as an immediate response to a system command. The completion of the response is not dependent on another system action or task.

6 **Job Status**: This message contains status information regarding the job or job step.

7 **Application Program/Processor**: This message is issued when a program is in problem mode.

8 **Out-of-Line Message**: This message is one of a group of messages to be displayed out of line. If the device support cannot print a message out of line, the code is ignored, and the message is printed in line with other messages.

9 **Request of the Operator**: This message is written in response to a request of the operator for information by the DEVSERV, MONITOR commands, and other operating system commands.

10 This message is issued in response to a **TRACK** command.

11 **Critical Eventual Action Required**: This message indicates that a critical event has occurred and must eventually be followed by an action. The message remains on the screen until the action is taken.

12 **Important Information**: This message contains important information that must be displayed at the console, but does not require any action in response.

13–16 Reserved.

### Message groups

A message group contains two or more messages that are displayed together in response to a specific command or error condition. The following example is a message group.

```plaintext
EZZ8453I jobtype STORAGE
EZZ8454I jobname STORAGE CURRENT MAXIMUM LIMIT
EZD2818I location
EZZ8455I storgetype current maximum limit
EZZ8459I DISPLAY TCP/IP STOR COMPLETED SUCCESSFULLY
```

In most cases, the "Explanation" section of the first message in the group contains an example of the group and information about all messages in the group. The message descriptions of members of the group refer back to the first message for complete information.
Chapter 2. EZB0xxxx messages

EZB0600I  message

Explanation
This message is received by the server when the dynamic allocation of a data set or the MVS enqueuing (ENQ) function was unsuccessful.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
S99Error

EZB0603I  offset

Explanation
This message indicates the offset at which the data string buffer is initialized.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DumpString

EZB0606I  buffer position
Explanation
This message indicates the position at which the data string buffer is initialized.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DumpString

EZB0611E   Data Set name “arguments” invalid.

Explanation
An argument declared in the hlq.LPD.CONFIG data set is incorrect. This message indicates the incorrect argument as declared by the user.

System action
LPD continues.

Operator response
None.

System programmer response
Correct the argument declared in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for information about the syntax rules for the hlq.LPD.CONFIG data set.

Module
LPD

Procedure name
ProcessOperands

EZB0612E   Data Set “arguments” not found.

Explanation
The argument declared in the hlq.LPD.CONFIG data set was not found. This message indicates the argument as declared by the user.
System action
LPD continues.

Operator response
None.

System programmer response
Correct the argument declared in the hlq.LPD.CONFIG data set and restart the program. See the z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
ProcessOperands

EZB0613E Data Set “dataset name” does not contain member “member name”.

Explanation
The data set name specified in the data set name parameter of the SEZAINST(LPSPROC) data set does not contain the indicated member.

System action
LPD ends.

Operator response
None.

System programmer response
Specify the correct data set name in the data set name parameter of the SEZAINST(LPSPROC) data set. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
ProcessOperands

EZB0614I IBM MVS LPD version version level

Explanation
This message indicates the current version and level of LPD for MVS.

System action
LPD continues.

Operator response
None.
System programmer response
None.

Module
LPD

Procedure name
ProcessVersionOption

EZB0615I   The option “option” is ambiguous. Use a longer abbreviation.

Explanation
An option specified in a parameter of the SEZAINST(LPSPROC) data set is incomplete. The option was abbreviated; however, the abbreviation is too short to distinguish between 2 correct options.

System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct option in the SEZAINST(LPSPROC) data set and restart the procedure. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
ProcessOptions

EZB0616I   The option “option” was not recognized.

Explanation
An option specified in a parameter of the SEZAINST(LPSPROC) data set is not recognized.

System action
LPD continues.

Operator response
None.

System programmer response
Check for the correct option parameter of the SEZAINST(LPSPROC) data set and verify that the option specified is supported on your system. See z/OS Communications Server: IP Configuration Reference for more information.
**EZB0617I** Use the TRACE, TYPE or VERSION options as needed.

**Explanation**
An incorrect option was specified in a parameter of the SEZAINST(LPSPROC) data set. This message indicates the correct options for this parameter. The following list provides the names and descriptions for these options:

**VERSION**
Displays the version number.

**TYPE**
Activates high-level trace facility in the LPD server. Significant events, such as the receipt of a job for printing, are recorded in the SYSOUT data set specified in your LPSPROC data set.

**TRACE**
Causes a detailed trace of activities in the LPD server to record in the SYSOUT data set specified in your LPSPROC data set. The detailed tracing can also be activated by the DEBUG statement in the configuration data set (hlq.LPD.CONFIG) and by the TRACE command of the SMSG interface.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Enter the correct option and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

**EZB0619I** Data set prefix not specified

**Explanation**
The prefix name of the TRACE name parameter of the SEZAINST(LPSPROC) data set was not specified. The parameter specifies the prefix of the configuration data set. The default is hlq. for the LPD.CONFIG data set.

**System action**
LPD continues.

**Operator response**
None.
System programmer response
Check for the correct data set prefix in the PREFIX name parameter of the SEZAINST(LPSPROC) data set. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
ProcessOptions

EZB0620I  Program error: Invalid option rc

Explanation
An error has occurred during processing of the configuration data set. This message indicates the return code received for this procedure.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Verify that the correct option parameters have been declared in the SEZAINST(LPSPROC) data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information. If the error persists, contact the IBM Software Support Center.

Module
LPD

Procedure name
ProcessOptions

EZB0621I  LPD starting with port port number

Explanation
This message indicates the port number on which the LPD connection was initiated.

System action
LPD continues.

Operator response
None.

System programmer response
None.
**Module**
LPD

**Procedure name**
ProcessArguments

---

**EZB0622E  message**

**Explanation**
The LPD is shutting down. See the text of the message for further explanation.

**System action**
LPD ends.

**Operator response**
None.

**System programmer response**
Check the message content for an indication of configuration problems. Reinitiate TCPIP, if required.

---

**Module**
LPD

**Procedure name**
Restore.

---

**EZB0624I  reason**

**Explanation**
The restore procedure, which resets the line printer daemon (LPD) counters to the default values and performs a cleanup routine, was initiated and LPD is ending. This message describes the reason for the procedure call and is not always an indication of a problem.

In the message text:

**reason**
The reason for the procedure call.

**System action**
LPD ends.

**Operator response**
Contact the system programmer if the message indicates a problem.

**System programmer response**
If this message indicates a problem, check for other messages in the LPD log that precede this one, such as EZB0622E and EZB0623I. Use the information from these other messages to determine what action to take and restart LPD.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: LPD

Module
LPD

Routing code
Not applicable.

Descriptor code
Not applicable.

Example
Not applicable.

**EZB0623I  \( \text{errmsg (msgnum)} \)**

Explanation
This message indicates the return code received by the Restore procedure. This procedure initiates a cleanup routine and ends TCPIP processing. This message is displayed with EZB0622I.

\( \text{errmsg} \) is the text of the message that describes the error.

\( \text{msgnum} \) is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in \( \text{errmsg} \). For more information about this message, see message **EZAmmsgnum** in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
LPD ends.

Operator response
None.

System programmer response
Respond as indicated by the message **EZAmmsgnum**.

Module
LPD

Procedure name
Restore

**EZB0625E**  Out of storage for connections!
**Explanation**
There is not enough storage allocated for a connection to complete.

**System action**
The program ends abnormally.

**Operator response**
None.

**System programmer response**
Allocate more storage for connections.

**Module**
LPD

**Procedure name**
AllocConnection

---

**EZB0626I**  **Allocated ConnectionBlock at address**

**Explanation**
A connection block was allocated at the indicated IP address. A connection block is used to build a connection record.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
AllocConnection

---

**EZB0627I**  **Passive open on port port number**

**Explanation**
A passive connection opening was established on the indicated port number.

**System action**
LPD continues.
Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
GetNewConnection

EZB0628I  Allocated PrinterBlock at address

Explanation
A printer block was allocated at the indicated IP address.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
AllocPrinter

EZB0629I  printer name added.

Explanation
This message indicates the name of the printer that was allocated for the LPD server.

System action
LPD continues.

Operator response
None.

System programmer response
None.
Module
LPD

Procedure name
AllocPrinter

**EZB0630I**  Spool allocate RC *rc*, Class *default class*, DEST *NJEdest*, ID *NJE ID*, OUTPUT *default spool*

**EZB0631I**  *string*

**Explanation**
The Network Job Entry system (NJE) is being used as the remote printing application for LPD. This message indicates the SYSOUT class, the name of the NJE node, the device user ID, the output name, and the value of the data buffer.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

---

Module
LPD

Procedure name
CompletePrinter

**EZB0632I**  *string*

**Explanation**
This message indicates the size of the data buffer.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.
Use DEST and IDENTIFIER for MVS.

Explanation

The DEST and IDENTIFIER parameters should be used for this MVS system. The DEST option sets the destination node. The default is the local node. The IDENTIFIER option specifies the device user ID. The default is SYSTEM.

System action

LPD continues.

Operator response

None.

System programmer response

None.

Module

LPD

Procedure name

CompletePrinter

Spool allocate RC spool rc, Class default class, OUTPUT default spool

Explanation

The Network Job Entry system (NJE) is being used as the remote printing application for LPD. This message indicates the SYSOUT class, the name of the NJE node, the device user ID, the output name, and the value of the data buffer.

System action

LPD continues.

Operator response

None.

System programmer response

None.

Module

LPD

Procedure name

CompletePrinter

string
Explanation
This message indicates the size of the data buffer.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
CompletePrinter

EZB0640I The errors reported above prevent startup of dataset name

Explanation
The indicated data set name could not be opened. See the previous messages, which provide more specific information about this error.

System action
LPD continues.

Operator response
None.

System programmer response
This message is displayed with more specific error messages. Respond as indicated by the previous messages.

Module
LPD

Procedure name
CompletePrinter

EZB0641I Service printer name defined with address address

Explanation
This message indicates the service name and the address space that were declared in the name parameter of the SERVICE statement of the hlq.LPD.CONFIG data set. The name parameter specifies a service where connections are accepted and acknowledged.

System action
LPD continues.
**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
CompletePrinter

**EZB0642I** Use a class which is one letter.

**Explanation**
The CLASS=class parameter of the service defined in the SERVICE statement of the hlq.LPD.CONFIG data set is incorrect. This parameter specifies the SYSOUT class. The default is A for printers and B for punches.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Specify the correct class in the CLASS=class parameter and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

**Module**
LPD

**Procedure name**
SetPrinterRSCSDefault

**EZB0643I** Use a class which is one letter or digit.

**Explanation**
The CLASS=class parameter of the service defined in the SERVICE statement of the hlq.LPD.CONFIG data set is incorrect. This parameter specifies the SYSOUT class. The default is A for printers and B for punches.

**System action**
LPD continues.

**Operator response**
None.
**System programmer response**

Specify the correct class in the CLASS=class parameter and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

**Module**

LPD

**Procedure name**

SetPrinterRSCSDefault

**EZB0644I** Use a number after the PRIORITY keyword.

**Explanation**

An unexpected character was received after the PRIORITY=priority parameter of the service defined in the SERVICE statement of the hlq.LPD.CONFIG data set. The PRIORITY parameter specifies the transmission priority. The default is 50.

**System action**

LPD continues.

**Operator response**

None.

**System programmer response**

Correct the entry declared in the PRIORITY=priority parameter and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

**Module**

LPD

**Procedure name**

SetPrinterRSCSDefault

**EZB0645I** SPOOL only valid for PRINTER or PUNCH

**Explanation**

SPOOL is valid for the PRINTER or PUNCH services only. These services are specified in the SERVICE statement of the hlq.LPD.CONFIG data set.

**System action**

LPD continues.

**Operator response**

None.

**System programmer response**

Correct the parameters declared for the services specified in the SERVICE statement. See z/OS Communications Server: IP Configuration Reference for more information.
Module
LPD

Procedure name
SetPrinterRSCSDefault

**EZB0646I**  Could not spool *dataset name device*. Return code *rc*

**Explanation**
The attempt to spool to the data set defined in the *name* parameter of the SERVICE statement was unsuccessful. This message indicates the return code received after this procedure.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Check for the correct data set name specified in the *name* parameter of the SERVICE statement in the `hlq.LPD.CONFIG` data set and restart the program. See *z/OS* Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
SetPrinterRSCSDefault

**EZB0647I**  Only use CLASS, DEST, IDENTIFIER, OTHERS, PRIORITY, SPOOL or TAG as qualifiers.

**Explanation**
An incorrect parameter was specified in the RSCS statement of the `hlq.LPD.CONFIG` data set. The following list provides a description for these parameters:

**CLASS**
The SYSOUT class. The default is A for printers and B for punches.

**DEST**
Specifies the destination node ID. The default is the node on which LPSERVE is running.

**IDENTIFIER**
Specifies the device user ID. The default is SYSTEM.

**OTHERS**
The option is ignored by MVS LPSERVE.

**PRIORITY**
Specifies the transmission priority. The default is 50.

**SPOOL**
Supplies the operands for the CP SPOOL command that will be used on the virtual printer or punch that is defined for each new job.
TAG
Supplies the operands for the CP TAG command that will be used on the virtual printer or punch that is defined for each new job.

System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct parameter in the RSCS statement of the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
SetPrinterRSCSDefault

EZB0648I Use a class which is one letter.

Explanation
The CLASS option of the NJE parameter for the SERVICE statement in the hlq.LPD.CONFIG data set is not correct. The default is A for printers and B for punches.

System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct CLASS option for the NJE parameter in the SERVICE statement and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
SetPrinterRSCSDefault

EZB0649I Use a class which is one letter or digit.

Explanation
The CLASS option of the NJE parameter for the SERVICE statement in the hlq.LPD.CONFIG data set is not correct. The default is A for printers and B for punches.
**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Specify the correct CLASS option for the NJE parameter in the SERVICE statement and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

**Module**
LPD

**Procedure name**
SetPrinterRSCSDefault

**EZB0650I** SPOOL only valid for PRINTER or PUNCH

**Explanation**
SPOOL is valid for the PRINTER or PUNCH services only. These services are specified in the SERVICE statement of the hlq.LPD.CONFIG data set.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Correct the parameters declared for the services specified in the SERVICE statement. See z/OS Communications Server: IP Configuration Reference for more information.

**Module**
LPD

**Procedure name**
SetPrinterRSCSDefault

**EZB0651I** Could not spool dataset name device. Return code rc

**Explanation**
The attempt to spool to the data set defined in the name parameter of the SERVICE statement was unsuccessful. This message indicates the return code received after this procedure.

**System action**
LPD continues.
Operator response
None.

System programmer response
Check for the correct data set name specified in the name parameter of the SERVICE statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
SetPrinterRSCSDefault

EZB0652I  Only use CLASS, DEST, SPOOL or TAG as qualifiers.

Explanation
An incorrect parameter was specified in the LOCAL statement of the hlq.LPD.CONFIG data set. The following list provides a description for these parameters:

CLASS
   The SYSOUT class. The default is A for printers and B for punches.
DEST
   Specifies the destination node ID. The default is the node on which LPSERVE is running.
SPOOL
   Supplies the operands for the CP SPOOL command that will be used on the virtual printer or punch that is defined for each new job.
TAG
   Supplies the operands for the CP TAG command that will be used on the virtual printer or punch that is defined for each new job.

System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct parameter in the RSCS statement of the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
SetPrinterRSCSDefault

EZB0653I  The keyword "keyword" is not a keyword.
Explanation
An incorrect parameter was declared in the SMTP statement of the hlq.LPD.CONFIG data set. The SMTP statement specifies the SMTP server name, CLASS, and DEST options.

System action
LPD continues. Failed job will not work unless the SMTP statement is defined and LPD is restarted.

Operator response
None.

System programmer response
Correct the parameter declared in the SMTP statement of the hlq.LPD.CONFIG data set, and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
ProcessSMTPOptions

EZB0654I  The keyword “keyword” is too short. Use a longer abbreviation.

Explanation
The indicated option was abbreviated. However, the abbreviation is too short to distinguish between 2 correct options.

System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct option and restart the procedure. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
ProcessLocalandRCSCOptions

EZB0655I  Use the SMTP service machine name after “SMTP”.

Explanation
An incorrect parameter was specified in the SMTP statement of the hlq.LPD.CONFIG data set. This statement specifies the SMTP server name, CLASS, and DEST options.
System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct name in the server_name parameter of the SMTP statement of the hlq.LPD.CONFIG data set and restart the program. If this parameter is omitted, the default is SMTP. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
ProcessSMTPOptions

EZB0656I The keyword “keyword” is not a keyword.

Explanation
An incorrect parameter was declared in the SMTP statement of the hlq.LPD.CONFIG data set. The SMTP statement specifies the SMTP server name, CLASS, and DEST options.

System action
LPD continues.

Operator response
None.

System programmer response
Correct the parameter declared in the SMTP statement of the hlq.LPD.CONFIG data set, and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
ProcessSMTPOptions

EZB0657I The keyword “keyword” is too short. Use a longer abbreviation.

Explanation
A parameter of the SMTP statement in the hlq.LPD.CONFIG data set is incorrect. This parameter was abbreviated; however, the abbreviation is too short to distinguish between two correct parameters.

System action
LPD continues.
Operator response
None.

System programmer response
Specify the correct parameter in the SMTP statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
ProcessSMTPOptions

EZB0658I Use a class which is one letter.

Explanation
The class specified in the CLASS=class parameter of the SMTP statement of the hlq.LPD.CONFIG data set is incorrect. This parameter specifies the SYSOUT class. The default is A for printers and B for punches. Valid values for this parameter are A through Z.

System action
LPD continues.

Operator response
None.

System programmer response
Correct the class declared in the CLASS=class parameter of the SMTP statement in the hlq.LPD.CONFIG data set and restart the program.

Module
LPD

Procedure name
ProcessSMTPOptions

EZB0659I Use a class which is one letter or digit.

Explanation
The class specified in the CLASS=class parameter of the SMTP statement of the hlq.LPD.CONFIG data set is incorrect. This parameter specifies the SYSOUT class. The default is A for printers and B for punches. Valid values for this parameter are A through Z and 0 through 9.

System action
LPD continues.

Operator response
None.
**System programmer response**
Correct the class declared in the CLASS=class parameter of the SMTP statement in the hlq.LPD.CONFIG data set and restart the program.

**Module**
LPD

**Procedure name**
ProcessSMTPOptions

**EZB0660I** Only use CLASS, DEST, IDENTIFIER or OTHERS as qualifiers.

**Explanation**
A parameter of the SMTP statement in the hlq.LPD.CONFIG data set is incorrect. This message indicates the correct parameters for this statement.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Specify the correct parameter for the SMTP statement of the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

**Module**
LPD

**Procedure name**
ProcessSMTPOptions

**EZB0665I** Cannot open configuration file.

**Explanation**
The configuration data set could not be opened.

**System action**
LPD ends.

**Operator response**
None.

**System programmer response**
Verify that the name of the configuration data set was specified correctly. Verify that the data set is available to the server. Verify that the server has authority to access the data set.
Module
LPD

Procedure name
PreparePrinters

EZB0666I Use option after you have defined a SERVICE.

Explanation
A parameter of the SERVICE statement in the hlq.LPD.CONFIG data set was not found. The SERVICE statement specifies a service for which connections are accepted and acknowledged. The following provides a description of the valid parameters for the SERVICE statement:

name
The service name must be one to eight characters in length. Only characters permitted in MVS data set names are valid. This value is case-sensitive.

PRINTER
Specifies that the service is to a printer.

PUNCH
Specifies that the service is to a punch device.

NONE
Specifies that the service is not currently in use.

System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct parameter for in the SERVICE statement of the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0667I The keyword "keyword" is ambiguous. Use a longer abbreviation.

Explanation
A statement of the hlq.LPD.CONFIG data set is incorrect. An abbreviation was used for this statement; however, the abbreviation is too short to distinguish between two correct statements.

System action
LPD continues.

Operator response
None.
System programmer response
Use the correct statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0668I The keyword “keyword” was not recognized.

Explanation
A statement in the hlq.LPD.CONFIG data set is incorrect.

System action
LPD continues.

Operator response
None.

System programmer response
Correct the statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0669I Use an integer after “DISK or UNIT”.

Explanation
A number is expected after the DISK or UNIT statements of the hlq.LPD.CONFIG data set.

System action
LPD halts.

Operator response
None.

System programmer response
Specify the correct value in the parameters of the DISK or UNIT statement of the hlq.LPD.CONFIG data set and restart the program.

Module
LPD
**Procedure name**
PreparePrinters

**EZB0670I** Use either START or END after “EXIT”.

**Explanation**
Specify the START or END parameter with the EXIT statement of the hlq.LPD.CONFIG data set. The following provides a description for these parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>START</td>
<td>Specifies that the program is invoked after allocating and opening the output data set, but before anything is written to the data set.</td>
</tr>
<tr>
<td>END</td>
<td>Specifies that the program is invoked just before closing the output data set.</td>
</tr>
</tbody>
</table>

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Use the correct parameter with the EXIT statement of the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

**Module**
LPD

**Procedure name**
PreparePrinters

**EZB0671I** Use either START or END after “EXIT”.

**Explanation**
An incorrect parameter was specified in the EXIT statement of the hlq.LPD.CONFIG data set. The following provides a description for these parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>START</td>
<td>Specifies that the program is invoked after allocating and opening the output data set, but before anything is written to the data set.</td>
</tr>
<tr>
<td>END</td>
<td>Specifies that the program is invoked just before closing the output data set.</td>
</tr>
</tbody>
</table>

**System action**
LPD continues.

**Operator response**
None.
System programmer response

Use the correct parameter with the EXIT statement of the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0672I Use name after type of EXIT.

Explanation
The program name specified in the program parameter of the EXIT statement in the hlq.LPD.CONFIG data set is incorrect or not found. This parameter specifies the name of the program to be invoked after allocating and opening, but before closing, an output data set.

System action
LPD continues.

Operator response
None.

System programmer response

Specify the correct program name in the program parameter of the EXIT statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0673I Could not load EXIT program name.

Explanation
The program name specified in the program parameter of the EXIT statement in the hlq.LPD.CONFIG data set could not be accessed. This parameter specifies the name of the program to be invoked after allocating and opening, but before closing, an output data set. This message is displayed when the START parameter was specified with the EXIT statement.

System action
LPD continues.

Operator response
None.
EZB0674I  Could not load EXIT program name.

Explanation
The program name specified in the program parameter of the EXIT statement in the hlq.LPD.CONFIG data set could not be accessed. This parameter specifies the name of the program to be invoked after allocating and opening, but before closing, an output data set. This message is displayed when the END parameter was specified with the EXIT statement.

System action
LPD continues.

Operator response
None.

System programmer response
Verify that the correct program name was specified in the program parameter of the EXIT statement in the hlq.LPD.CONFIG data set and restart the program.

The library containing the program should be in the system's link list, (LNKLSTxx) or a STEPLIB definition can be used if the library is APF authorized.

Module
LPD

Procedure name
PreparePrinters

EZB0675I  Use either MAIL or DISCARD after “FAILEDJOB”.

Explanation
A parameter specified in the FAILEDJOB statement of the hlq.LPD.CONFIG data is incorrect or not found. This statement specifies whether a notice of unsuccessful jobs should be mailed to users or a job is discarded without notification. The following provides a description for these parameters:

MAIL
- Specifies that notices of unsuccessful jobs are mailed to users.

DISCARD
- Specifies that unsuccessful jobs are discarded without notice.
System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct parameter in the FAILEDJOB statement of the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0676I Use an integer after “LINESIZE”.

Explanation
The value specified in the length parameter of the LINESIZE statement in the hlq.LPD.CONFIG data set was not found. This parameter specifies the number of characters in a line on a page. Lines longer than this number are truncated. The default is 132.

System action
LPD continues.

Operator response
None.

System programmer response
Correct the value declared in the length parameter of the LINESIZE statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0677I Use an integer after “LINESIZE”

Explanation
The value specified in the length parameter of the LINESIZE statement in the hlq.LPD.CONFIG data set is incorrect. This parameter specifies the number of characters in a line on a page. Lines longer than this number are truncated. The default is 132.

System action
LPD continues.
Operator response
None.

System programmer response
Correct the value declared in the length parameter of the LINESIZE statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZR0678I Use only one of LOCAL, NJE, RSCS, and REMOTE.

Explanation
The destination type parameter for the SERVICE statement in the hlq.LPD.CONFIG data set is not correct. The following list provides a description of the valid destination types for this parameter:

- **LOCAL**: Specifies that the data sets are written to the local MVS printer or punch.
- **NJE**: Specifies that the data sets are delivered to the Network Job Entry (NJE) system.
- **RSCS**: Specifies that the data sets are delivered to the Remote Spooling Communications Subsystem (RSCS).
- **REMOTE**: Specifies that the data sets are forwarded to a remote printer.

System action
LPD continues.

Operator response
None.

System programmer response
Correct the destination type parameter for the SERVICE statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZR0679I Allocated ObeyBlock at address

Explanation
The OBEY statement for the hlq.LPD.CONFIG data set has been received. This statement specifies user IDs authorized to use the SMSG interface provided with LPD. This message indicates that an address has been
allocated an ObeyBlock, or authorization to use the SMSG interface. See z/OS Communications Server: IP Configuration Reference for more information.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
PreparePrinters

EZB0680I Use an integer after “PAGESIZE”.

Explanation
The value specified in the lines parameter of the PAGESIZE statement in the hlq.LPD.CONFIG data set was not found. This parameter specifies the number of lines on a page. The default is 60.

System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct value in the lines parameter of the PAGESIZE statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0681I Use an integer after “PAGESIZE”.

Explanation
The value specified in the lines parameter of the PAGESIZE statement in the hlq.LPD.CONFIG data set is incorrect. This parameter specifies the number of lines on a page. The default is 60.
Operator response
None.

System programmer response
Specify the correct value in the `lines` parameter of the PAGESIZE statement in the `hlq.LPD.CONFIG` data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0682I Use a printer@hostname after “REMOTE”.

Explanation
The destination specified in the `printer@hostname` parameter of the REMOTE statement in the `hlq.LPD.CONFIG` data set was not found. This parameter indicates the destination printer at a specified IP host. This can be an IP name or an IP address.

System action
LPD continues.

Operator response
None.

System programmer response
Check for the correct entry in the `printer@hostname` parameter of the REMOTE statement in the `hlq.LPD.CONFIG` data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0683I Use a printer@hostname after “REMOTE”.

Explanation
The destination specified in the `printer@hostname` parameter of the REMOTE statement in the `hlq.LPD.CONFIG` data set is incorrect. This parameter indicates the destination printer at a specified IP host. This can be an IP name or an IP address.

System action
LPD continues.

Operator response
None.
System programmer response

Check for the correct entry in the printer@hostname parameter of the REMOTE statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module

LPD

Procedure name

PreparePrinters

EZB0684I Cannot reach destination

Explanation

The destination specified in the printer@hostname parameter of the REMOTE statement in the hlq.LPD.CONFIG data set could not be reached. This parameter indicates the destination printer at a specified IP host. This can be an IP name or an IP address.

System action

LPD continues.

Operator response

None.

System programmer response

Check for the correct destination in the printer@hostname parameter of the REMOTE statement, and for the correct service name specified in the name parameter of the SERVICE statement in the hlq.LPD.CONFIG data set, and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module

LPD

Procedure name

PreparePrinters

EZB0685I Use only one of LOCAL, NJE, RSCS, and REMOTE.

Explanation

The destination type parameter specified for the SERVICE statement in the hlq.LPD.CONFIG data set is not correct. The following list provides a description of the valid service names for this parameter:

LOCAL
  Specifies that the data sets are written to the local MVS printer or punch.

NJE
  Specifies that the data sets are delivered to the Network Job Entry (NJE) system.

RSCS
  Specifies that the data sets are delivered to the Remote Spooling Communications Subsystem (RSCS).

REMOTE
  Specifies that the data sets are forwarded to a remote printer.
System action
LPD continues.

Operator response
None.

System programmer response
Correct the destination type parameter declared for the SERVICE statement in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0686I  Host “host” resolved to address. Printer name is “printer name”.

Explanation
This message indicates the host name, IP address, and the printer name to which the destination address was resolved. The destination address is specified in the printer@hostname parameter of the REMOTE statement in the hlq.LPD.CONFIG data set.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
PreparePrinters

EZB0687I  Use only one of LOCAL, NJE, RSCS, and REMOTE.

Explanation
The service name specified in the name parameter of the SERVICE statement in the hlq.LPD.CONFIG data set is incorrect. The following list provides a description of the valid service names for this parameter:

LOCAL
  Specifies that the data sets are written to the local MVS printer or punch.

NJE
  Specifies that the data sets are delivered to the Network Job Entry (NJE) system.
RSCS
   Specifies that the data sets are delivered to the Remote Spooling Communications Subsystem (RSCS).

REMOTE
   Specifies that the data sets are forwarded to a remote printer.

System action
LPD continues.

Operator response
None.

System programmer response
Correct the service name declared in the name parameter of the SERVICE parameter in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0688I  Use a printer name and type after “SERVICE”.

Explanation
The service name specified in the name parameter of the SERVICE statement in the hlq.LPD.CONFIG data set was not found. The SERVICE statement specifies the service name for which connections are accepted and acknowledged.

System action
LPD continues.

Operator response
None.

System programmer response
Check for the correct name specified in the name parameter of the SERVICE statement and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0689I  The service “service name” has been described more than once.

Explanation
The service name specified in the name parameter of the SERVICE statement in the hlq.LPD.CONFIG data set is a duplicate. This statement specifies a service for which connections are accepted and acknowledged.
System action
LPD continues.

Operator response
None.

System programmer response
Check for the correct name specified in the name parameter of the SERVICE statement and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0690I Use a printer name and type after “SERVICE”.

Explanation
The service name specified in the name parameter of the SERVICE statement in the hlq.LPD.CONFIG data set was not found. This statement specifies a service for which connections are accepted and acknowledged.

System action
LPD continues.

Operator response
None.

System programmer response
Check for the correct name specified in the name parameter of the SERVICE statement and restart the program.

Module
LPD

Procedure name
PreparePrinters

EZB0691I Use “PRINTER” or “PUNCH” as a SERVICE type.

Explanation
The parameter specified in the SERVICE statement of the hlq.LPD.CONFIG data set is incorrect. This statement indicates a service for which connections are accepted and acknowledged. The following provides a description of the valid parameters for this statement:

name
The service name must be one to eight characters in length. Only characters permitted in MVS data set names are valid. This value is case-sensitive.

PRINTER
Specifies that the service is to a printer.
PUNCH
  Specifies that the service is to a punch device.

NONE
  Specifies that the service is not currently in use.

System action
LPD continues.

Operator response
None.

System programmer response
Verify that the correct parameter is specified in the SERVICE statement and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters.

EZB0692I Use a Volume Serial after “VOLUME”.

Explanation
An incorrect value was declared in the VOLUME statement of the hlq.LPD.CONFIG data set.

System action
LPD continues.

Operator response
None.

System programmer response
Check for the correct entry specified in the VOLUME statement of the hlq.LPD.CONFIG data set and restart the program. The correct length of this parameter is six characters.

Module
LPD

Procedure name
PreparePrinters

EZB0693I Use a Table Name after “TRANSLATETABLE”.

Explanation
The name specified in the name parameter of the TRANSLATETABLE statement in the hlq.LPD.CONFIG data set was not found. This parameter specifies the name of the translation table. If a DBCS conversion parameter is specified, name is used to determine which DBCS table to load.
**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Verify that the correct name was specified in the name parameter of the TRANSLATETABLE statement and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

**Module**
LPD

**Procedure name**
PreparePrinters

**EZB0694I** Could not load Translate Table.

**Explanation**
The translation table name specified in the name parameter of the TRANSLATETABLE statement in the hlq.LPD.CONFIG data set could not be loaded. This statement specifies the translation table to be used by the client and is found in the name.TCPXLBIN data set.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Verify that the correct translate table name is specified in the name parameter of the TRANSLATETABLE statement, verify that the name.TCPXLBIN data set is available to the server, and restart the program. If a DBCS conversion parameter is specified in the SMTP.SMTP.CONFIG data set, name is used to determine which DBCS translation table to load.

**Module**
LPD

**Procedure name**
PreparePrinters

**EZB0695I** Could not load translate table with name - name for printer printer_name

**Explanation**
The Line Printer Daemon (LPD) attempted to load an SBCS translation table corresponding to the name provided by the TRANSLATETABLE or XLATETABLE statement in the hlq.LPD.CONFIG data set. This statement specifies the translation table to be used by the remote client for SBCS ASCII to EBCDIC translations.

All data sets in the search order hierarchy for the required translate table data set either do not exist, or do not contain data in the required format for SBCS binary translate tables.
name is an input string that becomes part of the translation table data set name (for example, 
hlq.name.TCPXBLIN).

printer_name is the name of the printer service.

System action
LPD continues, however the printer service is unavailable.

Operator response
Notify the system programmer.

System programmer response
Configure a valid SBCS binary translate table data set in the search order hierarchy for the required SBCS translation table. See the z/OS Communications Server: IP Configuration Reference for more information about using translation tables including search order hierarchy and customization.

Module
LPD

Procedure name
PreparePrinters

EZB0696I  Program error: Invalid option option

Explanation
An incorrect statement was specified in the hlq.LPD.CONFIG data set. This message indicates the statement as declared in this data set.

System action
TCPIP ends.

Operator response
None.

System programmer response
Restart TCPIP, correct the statement specified in the hlq.LPD.CONFIG data set and restart the program. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
PreparePrinters

EZB0697I  ...End of Printer chain...

Explanation
The hlq.LPD.CONFIG data set statements have been processed to build the control tables representing the supported printers.
System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
PreparePrinters

EZB0698I InitEmulation failed

Explanation
The procedure InitEmulation, which starts the process to allow commands to run in non-EC mode machines, was not successful.

System action
TCPIP ends.

Operator response
None.

System programmer response
Run a 3270 type terminal emulator or use a 3270 type display station and restart TCPIP.

Module
LPD

Procedure name
PrepareTCP

EZB0699I Starting TCP/IP service connection

Explanation
A connection was initiated to the TCP/IP services.

System action
LPD continues.

Operator response
None.
System programmer response
None.

Module
LPD

Procedure name
PrepareTCP

EZB0700I  BeginTcpip: errmsg (msgnum)

Explanation
The procedure BeginTcpip, which informs the TCPIP address space that you want to start using its services, was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
TCPIP ends.

Operator response
None.

System programmer response
Respond as indicated by the message EZAmsgnum.

Module
LPD

Procedure name
PrepareTCP

EZB0701I  TCP/IP turned on.

EZB0702I  Host “host ID” Domain “domain ID” TCP/IP Service Machine “service ID”

Explanation
TCPIP services have been initiated for the indicated host, domain, and printer IDs.

System action
LPD continues.

Operator response
None.
**System programmer response**
None.

**Module**
LPD

**Procedure name**
PrepareTCP

---

**EZB0703I**  
**FSEND failed** errmsg (msgnum)

**Explanation**
The procedure FSEND, which sends data on a TCP connection, was unsuccessful.

(errmsg is the text of the message that describes the error.

(msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Respond as indicated by the message EZAmsgnum.

---

**Module**
LPD

**Procedure name**
SendACK

---

**EZB0704I**  
**Abort issued for connection connection number**

**Explanation**
The procedure DoAbortConnection, which shuts down a specific connection, was initiated. This message indicates the connection number for which the TcpAbort procedure was started.

**System action**
The TCP connection ends.

**Operator response**
None.

**System programmer response**
Reinitiate the connection if required.
Module
LPD

Procedure name
DoAbortConnection

EZB0705I  date time

Explanation
This message indicates the current date and time.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
PrintTimeStamp

EZB0706I  Terminating connection connection message

Explanation
Because of an incorrect return code received from TCP, the connection ends. This message indicates the 
connection number and the reason for the termination. This message is displayed with EZB0705I.

System action
This connection ends.

Operator response
None.

System programmer response
Reinitiate the connection if required.

Module
LPD

Procedure name
TerminateConnection

EZB0707I  Adding “message line” to message.
Explanation
The indicated message line was added to the message input buffer.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoSendQueueList

EZB0708I  FSend of response sent

Explanation
The procedure FSend, which sends data on a TCP connection, was initiated.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoSendQueueList

EZB0710I  New command command code (no operands)

Explanation
This message indicates the command code that was received from the client.

System action
LPD continues.
Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoNewCommand

EZB0711I New command command code data “parameter”.

Explanation
This message indicates the command code and the additional operands that have been received from the client.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoNewCommand

EZB0712I Command rejected. Printer “printer” not recognized.

Explanation
A command from the client was received that contains an operand with an unrecognized printer ID.

System action
The LPD to LPD connection ends.

Operator response
Reinitiate the connection, check for the correct printer ID, and resubmit the command. See z/OS Communications Server: IP User's Guide and Commands for more information.

System programmer response
Assist the user as necessary.
Module
LPD

Procedure name
DoNewCommand

EZB0713I  Printer “print” not found

Explanation
The printer specified in the remote printing command is incorrect. This message indicates the printer ID as declared by the user.

System action
LPD continues.

Operator response
Check for the correct printer ID and resubmit the command. See z/OS Communications Server: IP User's Guide and Commands for more information.

System programmer response
Assist the user as necessary.

Module
LPD

Procedure name
DoNewCommand

EZB0716I  Job job ID comment printer name site

Explanation
This message indicates the job ID that was placed onto the queue for the designated printer.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
PrintJobLogLine

EZB0717E  Could not erase dataset name RC=rc
Explanation
The job submitted by the client could not be erased from the queue after processing. This message indicates the data set name and return code that was passed.

System action
LPD continues.

Operator response
None.

System programmer response
Check for error messages in the LPD log and trace.

Module
LPD

Procedure name
EraseFile

EZB0718E Could not open “queue name QUEUE”.

Explanation
The specified queue could not be opened.

System action
LPD continues.

Operator response
None.

System programmer response
This message should be preceded by more specific messages. Correct the errors indicated by the preceding messages.

Module
LPD

Procedure name
SavePrinterQueue

EZB0719I Allocated JobBlock at address

Explanation
The print job block received from the client was allocated at the indicated IP address.

System action
LPD continues.
Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
AllocJob

EJB0720I Site file “name SITE” record 1 unreadable.

Explanation
The first record of the HOSTS.SITEINFO data set could not be read.

System action
LPD continues.

Operator response
None.

System programmer response
Verify that the HOSTS.SITEINFO data set was generated and installed, and that the records have been entered using the correct record format. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
LoadSite

EJB0721I Site file “name SITE” record 2 unreadable.

Explanation
The second record of the HOSTS.SITEINFO data set could not be read.

System action
LPD continues.

Operator response
None.

System programmer response
Verify that the HOSTS.SITEINFO data set was generated and installed and that the records have been entered using the correct record format.
Module
LPD

Procedure name
LoadSite

EZB0722I   Could not open “name SITE”.

Explanation
The HOSTS.SITEINFO data set could not be opened. When making changes to the HOSTS.LOCAL data sets, you must generate and install new HOSTS.SITEINFO and HOSTS.ADDRINFO data sets. Use the MAKESITE statement as either a TSO command or a batch job to generate the new data sets.

System action
LPD continues.

Operator response
None.

System programmer response
Verify that the HOSTS.SITEINFO data set was generated and installed and that the records have been entered using the correct record format.

Module
LPD

Procedure name
LoadSite

EZB0723I   Allocated StepBlock at address

Explanation
A StepBlock was allocated at the indicated IP address.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
AllocStep
Could not open control file “site name job ID printer name”. Job abandoned.

**Explanation**
The control data set specified in the LPR command received by the server could not be opened. This message indicates the site, the job ID, and the remote printer ID.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Verify that the correct data set name was specified in the LPR command and that the data set is available to the server and reissue the LPR command.

**Module**
LPD

**Procedure name**
ProcessControlFile

Reloading job job.

**Explanation**
The job ID, specified in the remote printing command, is reloaded.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
LoadJob

Could not open “site job ID”.

**Explanation**
The job ID specified in the LPR command received by the server could not be processed because the indicated site could not be opened. This message indicates the site and the job ID specified.
System action
LPD continues.

Operator response
Make sure the correct destination ID is used for its corresponding parameter and the correct job name is declared in the JOB parameter of the LPR command. See z/OS Communications Server: IP Configuration Reference for more information.

System programmer response
None.

Module
LPD

Procedure name
LoadJob

EZB0727I  Job job ID abandoned. Job file too short.

Explanation
The procedure readln, which reads the data set specified by the client on the LPR command, returned a nonzero return code.

System action
LPD continues.

Operator response
None.

System programmer response
Check for the correct job ID specified in the Job jobname parameter of the LPR command and submit the command again. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
LoadJob


Explanation
The procedure readln, which reads the data set specified by the client on the LPR command, returned a nonzero return code.

System action
LPD continues.
Operator response
None.

System programmer response
Check for the correct job ID specified in the Job jobname parameter of the LPR command and submit the command again. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
LoadJob


Explanation
The procedure readln, which reads the data set specified by the client on the LPR command, returned a nonzero return code.

System action
LPD continues.

Operator response
None.

System programmer response
Check for the correct job ID specified in the Job jobname parameter of the LPR command and submit the command again. See z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
LoadJob

EJB0730I  Job job ID abandoned. Not enough storage.

Explanation
The job ID specified in the jobname parameter of the LPR command received by the server could not be completed because of insufficient storage.

System action
LPD continues.

Operator response
Inform the system programmer about this message.
System programmer response
Check for the storage requirements needed to process the LPR command, allocate more storage, and reissue the command.

Module
LPD

Procedure name
LoadJob

EZB0731I Work Queue start

Explanation
The work queue was initiated.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
PrintWorkQueue

EZB0732I  job number job ID

Explanation
This message indicates the print job received from the client was placed onto the queue for the designated service.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD
**Procedure name**
PrintWorkQueue

**EZB0733I  Work Queue end**

**Explanation**
The work queue is empty. The print server returns to a passive wait state awaiting the next print request.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
PrintWorkQueue

**EZB0734I  Job jobnumber: added to work queue**

**Explanation**
TCPIP displays this message while tracing is on. The specified job number is displayed as it is added to the list of queued jobs.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
QueueJobWork

**EZB0735I  stepblock_address datasize action_code dataset name**

**Explanation**
TCPIP issues this message when tracing is on. The StepBlock address, data size, action code and data set name of the current job are displayed.
**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
DumpStepChain

**EZB0736I**  Could not open *name* QUEUE

**Explanation**
LPD could not open the specified data set that is queued for printing.

**System action**
LPD continues.

**Operator response**
Check the syntax of the specified data set name or verify that you have authority to print through the system programmer.

**System programmer response**
Assist the user as necessary.

**Module**
LPD

**Procedure name**
LoadPrinterQueue

**EZB0737I**  Reloading *dataset name* queue

**Explanation**
LPD issues this message while tracing is on. While attempting to queue the previous data set name, an error was detected. The data set name is reloaded into the list of queued data sets.

**System action**
LPD continues.

**Operator response**
None.
**System programmer response**
None.

**Module**
LPD

**Procedure name**
LoadPrinterQueue

---

**EJB0738I**  Ignoring job *number* from site *name* because there is no SITE file.

**Explanation**
TCPIP was unable to locate the site for the specified job number.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Make sure the site was defined using the MAKESITE command.

---

**Module**
LPD

**Procedure name**
LoadPrinterQueue

---

**EJB0739I**  Validating user *user*

**Explanation**
This message occurs while tracing is on. The userid of the submitter of the print job is validated.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

---
**Procedure name**
LoadPrinterQueue

**EZB0740I** Validation failed. RC rc

**Explanation**
TCPIP issues this message while tracing is on. While attempting to validate the password for the currently queued job an error was detected.

**System action**
LPD continues.

**Operator response**
Inform the system programmer of the error.

**System programmer response**
Assist the user as necessary.

**Module**
LPD

**Procedure name**
ValidateJob

**EZB0744I** address punch line

**Explanation**
LPD displays the address of the punch command text and the punch line number. The punch line text represents the information sent to SMTP from LPD.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
Punchline

**EZB0747E** Could not allocate SMTP Spool

**Explanation**
The SMTP device could not allocate the spool for the batch data set submitted using the SMTP command.
System action
LPD continues.

Operator response
Notify the system programmer.

System programmer response
Make sure the SMTP device is activated.

Module
LPD

Procedure name
SendFailingMail

EZB0748E  Could not open spool to SMTP. Return code was last error

Explanation
LPD could not spool a job to the indicated SMTP service. A nonzero return code was returned.

System action
LPD continues.

Operator response
Notify the system programmer.

System programmer response
Make sure the SMTP statement is updated to include accurate information for its parameters in the hlq.LPD.CONFIG data set. Also make sure that the MAIL parameter is used with the FAILEDJOB statement. For more information about the SMTP and FAILEDJOB statements, see z/OS Communications Server: IP Configuration Reference. Check the return code issued in this message and the return codes listed under SMTP in the z/OS Communications Server: IP and SNA Codes to further determine and correct the error.

Module
LPR

Procedure name
SendFailingMail

EZB0750E  Could not deallocate Spool File ddname Error code was rc

Explanation
The SMTP server was unable to deallocate the specified spool file after attempting to close the SMTP connection.

System action
LPD continues.
**Operator response**
Notify the system programmer.

**System programmer response**
Check the return code issued in this message and the return codes listed under SMTP in the z/OS Communications Server: IP and SNA Codes to determine and correct the error.

**Module**
LPR

**Procedure name**
SendFailingMail

**EZB0751I** Released StepBlock at address

**Explanation**
The SMTP server displays this message as the job at the specified address is deleted from storage.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
RemoveJobFiles

**EZB0753I** New subcommand command (no operands)

**Explanation**
This message is displayed while tracing is on. A new SMTP subcommand was detected by LPD with no optional operands.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.
Module
LPR

Procedure name
DoNewSubcommand

EZB0754I New subcommand command operands operands.

Explanation
This message is displayed while tracing is on. A new SMTP subcommand was detected by LPD. The new subcommand and optional operands are displayed.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
DoNewSubcommand

EZB0755I Released StepBlock at address

Explanation
This message occurs while tracing is on. This message is displayed when the previous subcommand is deleted from the stated storage area.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
DoNewSubcommand

EZB0756I Job number number is invalid.
Explanation
The LPD server was unable to queue the specified job number for processing. The connection was terminated.

System action
LPD continues.

Operator response
Resubmit the job using the LPR command.

System programmer response
Assist the user as necessary.

Module
LPD

Procedure name
DoNewSubcommand

EZB0757I  Duplicate file name "dataset name".

Explanation
This message indicates that the specified data set was previously recognized by LPD. The job is ignored and the next job is processed.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoNewSubcommand

EZB0759E  Failed to allocate block for jobnumber from site

Explanation
The service machine was unable to allocate enough storage to process the specified job at the specified site.

System action
LPD continues.
Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoNewSubCommand

EZB0760E   Failed to open “dataset name”.

Explanation
The server machine's attempt to open the stated data set failed. The connection is terminated.

System action
LPD continues.

Operator response
Notify the system programmer.

System programmer response
Check for preceding error message EZB0600I in the LPD log to see why the open failed.

Module
LPD

Procedure name
DoNewSubcommand

EZB0761E   Could not open data file “dataset name”. Job abandoned.

Explanation
The server machine issues this message after attempting to open the data file at the specified site with the specified file type. The job is abandoned.

System action
LPD continues.

Operator response
Notify the system programmer.

System programmer response
Make sure the specified site is defined by the MAKESITE command.
Module
LPR

Procedure name
DoSendStep

EZB0762I Sending subcommand command with an operand of operand

Explanation
LPD issues this message while tracing is on. LPD acknowledges sending the indicated subcommand with the indicated operand.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
DoSendJob

EZB0763I Closing connection connection

Explanation
This message is issued while tracing is on. The connection between the client and the LPD server has been closed.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoCloseConnection

EZB0764I ACK received on connection connection for job jobnumber in state jobstate
Explanation
This message is issued while tracing is on. LPD has received an acknowledgment of the specified connection. The current state of the job is also displayed.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoNewAck

EZB0765I ACK in unexpected job state jobstate

Explanation
This message is issued while tracing is on. LPD received an ACK from the remote host, but the job state was bad. See message EZB0764I for job and connection number.

System action
LPD continues.

Operator response
Check all physical ports and power switch to verify that the printer is ready for printing. If the problem persists, contact your hardware support personnel.

System programmer response
None.

Module
LPD

Procedure name
DoNewAck

EZB0766I NACK has value number

Explanation
This message is issued while tracing is on. The remote host refused to complete processing for this job.

System action
LPD continues.
Operator response
Check the printer and make sure it is active. If the problem persists, contact your hardware support personnel.

System programmer response
None.

Module
LPD

Procedure name
DoNewAck

EZB0767I    Timer cleared for connection connection

Explanation
This message is issued while tracing is on. LPD has successfully cleared the timer for the specified connection.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoNewData

EZB0768I    Ignoring Data delivered on connection connection

Explanation
This message is issued while tracing is on. The buffer has been exhausted. The data to be delivered is ignored.

System action
LPD continues.

Operator response
Notify the system programmer.

System programmer response
Increase the specified buffer size using the DATABUFFERPOOLSIZExx statement.
Module
LPD

Procedure name
DoNewData

EZB0769I  Job jobnumber removed from work queue

Explanation
This message is issued while tracing is on. LPD issues this message when the specified job number is removed from the list of queued jobs.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
RemoveJobWork

EZB0770I  Job jobnumber not found in printer chain.

Explanation
LPD issues this message when the specified job number is not found in the print queue.

System action
LPD halts.

Operator response
Notify the system programmer.

System programmer response
Check the LPD trace. Job may have already printed. Restart LPD.

Module
LPD

Procedure name
RemoveJobPrinter

EZB0771I  Released JobBlock at address

72  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Explanation
LPD issues this message while tracing is on. The storage space allocated for the job block at the specified address was released.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
FreeJob

**EZB0772I** End Connection connection for **errmsg (msgnum)**

Explanation
The specified LPD server connection has ended.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
LPD continues.

Operator response
None.

System programmer response
Respond as indicated by the message EZAmsgnum.

Module
LPD

Procedure name
DoEndConnection

**EZB0773I** Connection connection terminated for **errmsg (msgnum)**

Explanation
This message indicates that the specified connection has been terminated.
errmsg is the text of the message that describes the error.
msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Respond as indicated by the message EZAmsgnum.

**Module**
LPD

**Procedure name**
DoEndConnection

**EZB0774I** Connection ended abruptly.

**Explanation**
The LPD connection with the remote printer has terminated. LPD received a job and the connection state was not CONNCLOSING.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Review the trace and restart the job. If DEBUG is not on, define DEBUG in the LPD.CONFIG and restart the job.

**Module**
LPD

**Procedure name**
DoEndConnection

**EZB0775I** Released StepBlock at address

**Explanation**
This message is issued while tracing is on. The block at the specified address was freed of storage.

**System action**
LPD continues.
Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoEndConnection

**EZB0776I** Released StepBlock at address

Explanation
This message is issued while tracing is on. The storage space specified was released because there were no jobs awaiting processing.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoEndConnection

**EZB0777I** Released ConnectionBlock at address

Explanation
This message is issued while tracing is on. The storage space specified for the LPD connection was released because there were no jobs awaiting processing.

System action
LPD continues.

Operator response
None.

System programmer response
None.
Module
LPD

Procedure name
DoEndConnection

EZB0778S    LPD terminating because errmsg (msgnum)

Explanation
This message is displayed while tracing is on. The LPD server has terminated operation.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
LPD halts.

Operator response
Notify system programmer.

System programmer response
Respond as indicated by the message EZAmsgnum.

Module
LPD

Procedure name
DoEndConnection

EZB0779I    New connection state state on connection address with reason errmsg (msgnum)

Explanation
This message indicates that LPD has made a new connection state at the specified address.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
LPD continues.

Operator response
None.

System programmer response
Respond as indicated by the message EZAmsgnum.
Module
LPD

Procedure name
DoNewConnState

EZB0780I  Abort failed errmsg (msgnum)

Explanation
This message is issued as a result of the LPD's unsuccessful attempt to disconnect the TCP connection. 
errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in 
cerrmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
LPD continues.

Operator response
None.

System programmer response
Respond as indicated by the message EZAmsgnum.

Module
LPD

Procedure name
DoEndConnection

EZB0781I  Connection aborted because port number (number) is out of range.

Explanation
This message is issued while tracing is on. The LPD connection was aborted because the stated port number was 
not defined as a legitimate port. Legitimate ports range from 721 to 731, inclusive.

System action
LPD continues.

Operator response
Notify the system programmer.

System programmer response
Use the PORT statement to define the legitimate ports. For more information about the PORT statement, see the 
z/OS Communications Server: IP Configuration Reference.

Module
LPD
**Procedure name**  
DoNewConnState

**EZB0782I**  
Connection open. Reading command.

**Explanation**  
This message is issued while tracing is on. The LPD connection is open and now reading the submitted connection state command.

**System action**  
LPD continues.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
LPD

**Procedure name**  
DoNewConnState

**EZB0783I**  
Aborting Connection connection - Timed out

**Explanation**  
The specified connection was aborted because the timer expired.

**System action**  
LPD continues.

**Operator response**  
None.

**System programmer response**  
Assist the user as necessary. Use the TIMER command to set the timer to an appropriate time limit.

**Module**  
LPD

**Procedure name**  
DoConnectionTimeOut

**EZB0784I**  
Could not retrieve SMSG

**Explanation**  
LPD was unable to retrieve a queued Smsg. Therefore, the message could not be queued for processing.
System action
LPD continues.

Operator response
None.

System programmer response
See message EZB0800I.

Module
LPD

Procedure name
ProcessSMSG

EZB0785I Attempted SMSG from “user” ignored.

Explanation
The SMSG from the specified user was ignored. LPD was unable to locate the user ID in the OBEYFILE.

System action
LPD continues.

Operator response
Notify the system programmer.

System programmer response
Make sure the specified user ID is listed in the OBEYFILE using the OBEY command.

Module
LPD

Procedure name
ProcessSMSG

EZB0786I Command received “string”

Explanation
LPD issues this message when the specified special messages string is received.

System action
LPD continues.

Operator response
None.
System programmer response
None.

Module
LPD

Procedure name
ProcessSMSG

EZB0788I Command not understood.

Explanation
LPD does not recognize the command issued through the SMSG interface.

System action
LPD continues.

Operator response
Correct the syntax and reissue the command.

System programmer response
None.

Module
LPD

Procedure name
ProcessSMSG

EZB0789I GetNextNote with ShouldWait of number

Explanation
This message is issued while tracing is on. TCP is initializing processing procedures to retrieve the next queued notification. The ShouldWait function is set to either a true or false value depending on whether TCP is to wait for notification before the next one is available.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD
Procedure name
ProcessTCP

EZB0790I GetNextNote returns. Connection connection Notification notification

Explanation
This message is issued while tracing is on. The connection number and the notification status are returned upon successful completion of the GetNextNote procedure.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
ProcessTCP

EZB0791I New TCP notice arrived

Explanation
This message is issued while tracing is on. TCP has received the next queued notification.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
ProcessTCP

EZB0792I Connection: connection

Explanation
This message is issued while tracing is on. This message displays the connection number of the TCP connection.
**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
ProcessTCP

---

**EZB0793I Notification: notification**

**Explanation**
This message is issued while tracing is on. This message displays the notification status of the current TCP connection.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
ProcessTCP

---

**EZB0794I NewState: newstate**

**Explanation**
This message is issued while tracing is on and notification has changed. The new state is displayed in this message.

**System action**
LPD continues.

**Operator response**
None.
System programmer response
None.

Module
LPD

Procedure name
ProcessTCP

EZB0795I  ConnState: connectionstate

Explanation
This message is issued while tracing is on. If TCP does not detect a connection state change, this message is displayed.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
ProcessTCP

EZB0796I  BytesDelivered: number

Explanation
This message is issued while tracing is on. The number of data bytes delivered on the TCP connection are displayed.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD
**Procedure name**

ProcessTCP

**EZB0797I  SendTurnCode: errmsg (msgnum)**

**Explanation**

This message is issued while tracing is on. TCP failed in its attempt to send data on the TCP connection. `errmsg` is the text of the message that describes the error.

`msgnum` is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in `errmsg`. For more information about this message, see message EZA`msgnum` in the z/OS Communications Server: IP Messages Volume 1 (EZA).

**System action**

LPD continues.

**Operator response**

None.

**System programmer response**

Respond as indicated by the message EZA`msgnum`.

**Module**

LPD

**Procedure name**

ProcessTCP

**EZB0798I  Queueing job jobnumber**

**Explanation**

This message is issued while tracing is on. The specified job is queued for processing.

**System action**

LPD continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

LPD

**Procedure name**

ProcessTCP

**EZB0799I  Reading additional data on connection**
**Explanation**
This message is issued while tracing is on. Processing of the current job continues on the specified connection.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
ProcessTCP

**EZB0800I Ignoring TCP/IP notice notification on connection**

**Explanation**
This message is issued while tracing is on. The TCP connection has failed or was not recognized. Use the type identifier issued in this message and the z/OS Communications Server: IP Programmer's Guide and Reference to determine and correct the problem.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Assist the user as necessary.

**Module**
LPD

**Procedure name**
ProcessTCP

**EZB0801I Filter option not supported. Job abandoned.**

**Explanation**
This message could display for one of the following reasons:
- The option type of the filters parameter used with the LPR command is not supported by the line printer daemon.
- The option type of the filters parameter for the SERVICE statement in the hlq.LPD.CONFIG data set is not supported by the line printer daemon.
System action
LPD continues.

Operator response
Correct the option type of the filters parameter for the LPR command or in the SERVICE statement of the hlq.LPD.CONFIG data set. Reissue the job. See z/OS Communications Server: IP User's Guide and Commands for information about the LPR command or the z/OS Communications Server: IP Configuration Reference for information about the SERVICE statement.

System programmer response
None.

Module
LPD

Procedure name
DoStartStep

EZB0802E  Could not open data file dataset Job abandoned.

Explanation
LPD was unable to access the indicated data set. The print job is not completed.

System action
LPD continues.

Operator response
Reissue the print job with a valid data set name or PDS member. See z/OS Communications Server: IP User's Guide and Commands for more information about using the LPR command. None.

System programmer response
None.

Module
LPD

Procedure name
DoStepStart

EZB0803E  Could not define device for job. Unknown type type treated as “NONE”.

Explanation
LPD encountered a printer type defined in the SERVICE statement that was not valid or unknown.

System action
LPD continues.
Operator response
None.

System programmer response
Correct the device type parameter for the SERVICE statement in hlq.LPD.CONFIG data set. For more information about the SERVICE statement, see z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
GetJobDevice

EZB0804E printer kind is unknown address

Explanation
LPD was not able to recognize the route parameter for the indicated printer. The route parameter was defined to something other than remote, local or NJE.

System action
LPD continues.

Operator response
Notify the system programmer of this message.

System programmer response
Correct the route parameter for the SERVICE statement in the hlq.LPD.CONFIG data set. For information about the SERVICE statement, see z/OS Communications Server: IP Configuration Reference for more information.

Module
LPD

Procedure name
GetJobDevice

EZB0805E Could not allocate Spool Class class

Explanation
The LPD server was unable to set a spool file for the specified printer class as a result of buffer exhaustion. See messages EZB0806I and EZB0807I.

System action
LPD continues.

Operator response
None.
System programmer response
Resubmit the LPD command.

Module
LPD

Procedure name
GetJobDevice

EZB0806I  Copies copies, Font font, form form, output printer

Explanation
LPD issues this message when a nonzero return code is returned during spool file allocation. The number of copies, font size, form, and printer name are provided. See message EZB0807I.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
GetJobDevice

EZB0807I  bufferlength

Explanation
The length of the data buffer which failed while attempting to allocate a spool file is displayed.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD
**Procedure name**
GetJobDevice

**EZB0808I  bufferlength**

**Explanation**
The length of the alternate data buffer which failed while attempting to allocate a spool file is displayed.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
GetJobDevice

**EZB0809E  Could not open spool to destination  Return code was error**

**Explanation**
LPD was unable to open a spool file to the specified printer destination. A return code is passed.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Use the return code listed in this message and z/OS Communications Server: IP and SNA Codes.

**Module**
LPD

**Procedure name**
GetJobDevice

**EZB0810I  Spool printertype address for user. Return code was rc**

**Explanation**
LPD issues this message while tracing is on or a nonzero return code is displayed. The printer type, address, user, and return code are displayed with this message after the spool is allocated.
System action
LPD continues.

Operator response
None.

System programmer response
Check the meaning of the RC issued by CP spool “FOR” command.

Module
LPD

Procedure name
GetJobDevice

EZB0811E  printertype kind is unknown deviceaddress

Explanation
The LPR command has a destination defined that is not known. The destination is defined after the jobname parameter.

System action
LPD continues.

Operator response
Specify the correct destination in the jobname parameter and reissue the LPR command. For more information about the LPR command see the z/OS Communications Server: IP User's Guide and Commands.

System programmer response
None.

Module
LPD

Procedure name
GetJobDevice

EZB0812I  Spool TO address superseded by FOR user

Explanation
The SMTP spool to the default user address was superseded by the specified user ID.

System action
LPD continues.

Operator response
None.

90  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**EZB0813I   Spooling printer this way how**

**Explanation**
This message is displayed while tracing is on. LPD acknowledges the parameter indicated for the LPR command on the specified printer.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

---

**EZB0814E   Could not spool printer address type. Return code was rc**

**Explanation**
This message is issued while tracing is on. The LPD server was not able to spool the printer at the stated address with the specified type. A nonzero return code was returned.

**System action**
Printer not spooled.

**Operator response**
None.

**System programmer response**
Use the return code value to determine the error and reissue the command.

---

**Module**
LPD
**Procedure name**
GetJobDevice

**EZB0815I**  Tagging *printer* with *tag*

**Explanation**
This message is issued while tracing is on. The indicated RSCS tag has been assigned to the indicated printer.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
GetJobDevice

**EZB0816E**  Could not TAG *printer address tag* Return code was *rc*

**EZB0817I**  Response was *response*

**Explanation**
The LPD server was not able to assign the specified tag to the printer at the indicated address. A nonzero return code was passed.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Use the return code value and the response string provided with message EZB0817I to determine and correct the cause of the error.

**Module**
LPD

**Procedure name**
GetJobDevice

**EZB0819I**  Job *jobnumber* rescheduled -- no storage
**Explanation**
This message is issued while tracing is on. LPD was unable to allocate a connection for the specified job number. No storage block was available for a connection.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
DoStartSending

---

**EZB0820I**  Trying to open with local port *port*

**Explanation**
This message is issued while tracing is on. LPD is attempting to open the specified local port for communication.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
DoStartSending

---

**EZB0821I**  Job *jobnumber* abandoned -- Open failed or no ports

**Explanation**
This message is issued while tracing is on.

- LPD was unable to open a connection to the specified REMOTE Print Server (LPD) on another host. A TcpOpen error occurred.
- LPD was unable to open a local port to the specified REMOTE Print Server (LPD) on another host. No ports in the range 721–731 were available.

In the message text:
**jobnumber**
The number assigned by the LPR client to this job.

**System action**
LPD continues.

**Operator response**
Resubmit the print job when a connection to the REMOTE print server can be established.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
DoStartSending

**EZB0822I**  Sending command *number* with operand *address*

**Explanation**
This message is issued while tracing is on. LPD is sending the specified command to the specified printer address.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
DoOpenSending

**EZB0823I**  Sending ACK at end of data file on *connection* for job *jobnumber*

**Explanation**
This message is issued while tracing is on. The LPD server has read the data set on the specified connection for the specified job number and is sending an ACK.

**System action**
LPD continues.
Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
DoContinueSending

**EZB0824I  ProcessWork starting on job queue**

Explanation
This message is issued while tracing is on. LPD is now processing the next job queued.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
ProcessWork

**EZB0825I  Job jobnumber for printer dispatched in state state**

Explanation
This message is issued while tracing is on. The job number, printer, and print state are displayed.

System action
LPD continues.

Operator response
None.

System programmer response
None.
**Module**
LPD

**Procedure name**
ProcessWork

**EZB0826E  Job jobnumber (state) abandoned - Incorrect state.**

**Explanation**
LPD has terminated the processing of the specified job number because an incorrect job state was detected.

**System action**
LPD continues.

**Operator response**
Resubmit the job using the LPR command.

**System programmer response**
Make sure the failed job is mailed to the user using the MAIL command. For more information see the z/OS Communications Server: IP Programmer's Guide and Reference.

**Module**
LPD

**Procedure name**
ProcessWork

**EZB0827I  ProcessWork end with queue**

**Explanation**
This message is issued while tracing is on. LPD has finished processing the queued jobs.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
ProcessWork

**EZB0831I  IBM MVS LPD Version version on date at time**
**Explanation**
When the LPD server is initialized this message is displayed. The LPD version number, date, and starting time are displayed.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPD

**Procedure name**
LPD MAIN

---

**EZB0833I**  Could not get identity!

**Explanation**
The LPD server was unable to identify the user ID, host name, TCP service and domain name during its start up procedure. LPD does not start.

**System action**
LPD continues.

**Operator response**
None.

**System programmer response**
Make sure the user ID, host name, TCP service, and domain name have all been specified using the LPD command and its subcommands. For more information of the LPD command and its subcommands, see z/OS Communications Server: IP Configuration Reference for more information.

**Module**
LPD

**Procedure name**
LPD MAIN

---

**EZB0834I**  Ready

**Explanation**
LPD is initialized and ready for processing.
System action
LPD is initialized.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
LPD MAIN

EZB0835I Ignored data is string

Explanation
This message displays while tracing is on; it is preceded by EZB0768I. Data was delivered on the connection, but because the data is unrecognizable, it is ignored. The connection is closed.

System action
LPD continues.

Operator response
See EZB0768I.

System programmer response
See EZB0768I. Use the data from the message to determine if the LPD was sent bad data.

Module
LPD

Procedure name
DoNewData

EZB0850I Using Table for DBCS Translate: data set

Explanation
The DBCS translation table has been successfully loaded from the binary translation table data set in the search order hierarchy.

System action
LPD continues.

Operator response
None.
System programmer response
None.

Module
LPD

Procedure name
LoadDbcsTables

**EZB0851I**  Could not load DBCS Translate Table: *data set*

Explanation
DBCS conversion is configured for the LPD, but the required DBCS translation table could not be loaded.

System action
The program continues.

Operator response
None.

System programmer response
Configure a valid DBCS binary translate table data set in the search order hierarchy for the required DBCS translation table. See z/OS Communications Server: IP Configuration Reference for more information about loading and customizing DBCS translation tables.

Module
LPD

Procedure name
LoadDbcsTables

**EZB0852I**  Use a NLS option after "NLTRANSLATE".

Explanation
An NLS option is expected after the NLTRANSLATE keyword of the SERVICE statement of the *hlq*.LPD.CONFIG data set.

System action
LPD continues.

Operator response
None.

System programmer response
Specify the correct value in the parameter of NLTRANSLATE keyword of the SERVICE statement of the *hlq*.LPD.CONFIG data set and restart the program.
Module
LPD

Procedure name
ProcessNisOptions

EZB0853E  Use an integer after JOBPACING.

Explanation
A number is expected after the JOBPACING statement in the hlq.LPD.CONFIG data set.

System action
LPD continues using the default value.

Operator response
None.

System programmer response
Specify the correct value for the parameter in the JOBPACING statement in the hlq.LPD.CONFIG data set and restart the program.

Module
LPD

Procedure name
PreparePrinters

EZB0854E  Use an integer after STEPLIMIT.

Explanation
A number is expected after the STEPLIMIT statement in the hlq.LPD.CONFIG data set.

System action
LPD continues using in the default value.

Operator response
None.

System programmer response
Specify the correct value for the parameter in the STEPLIMIT statement in the hlq.LPD.CONFIG data set and restart the program.

Module
LPD

Procedure name
PreparePrinters

EZB0855I  Loaded translation table from dataset for printer printer_name

100  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Explanation
The Line Printer Daemon (LPD) loaded the following SBCS translation table corresponding to the name provided by the TRANSLATETABLE or XLATETABLE statement in the hlq.LPD.CONFIG data set. This statement specifies the translation table to be used by the remote client for SBCS ASCII to EBCDIC translations.

**dataset** is the fully qualified data set name.

**printer_name** is the name of the printer service.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
PreparePrinters

**EZB0856I**  Loaded translation table from **dataset**

Explanation
The Line Printer Daemon (LPD) loaded the SBCS translation table to be used if the printer service does not have TRANSLATETABLE or XLATETABLE statements specified.

**dataset** is the fully qualified data set name.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
ProcessOptions

**EZB0857I**  Using hardcoded translation tables.
Explanation
Hardcoded translation tables will be used because the Line Printer Daemon (LPD) was unable to load an SBCS translation table using the data set names of *jobname*.STANDARD.TCPXLBIN or *hlq*.STANDARD.TCPXLBIN. These tables are equivalent to *hlq*.STANDARD.TCPXLBIN, which are shipped with the product.

See z/OS Communications Server: IP Configuration Reference for more information about using translation tables including search order hierarchy and customization.

System action
LPD continues.

Operator response
None.

System programmer response
None.

Module
LPD

Procedure name
ProcessOptions

EZB0900I command name version version

Explanation
Displays the line printer command used and the version level of the program. This message is displayed when you use the version parameter with the LPR, LPQ, LPRM, and LPRSET commands.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR, LPQ, LPRM, LPRSET

Procedure name
ProcessVersionOption

EZB0901E The option option is ambiguous. Use a longer abbreviation.

Explanation
The abbreviated option submitted at the command line is ambiguous.
**System action**
The command is terminated.

**Operator response**
Reissue the required option with a longer abbreviation at the LPR command line. For a list of valid abbreviations using the LPR command, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
None.

**Module**
LPR, LPRM

**Procedure name**
ProcessOptions

**EZB0902E  Use a host name after HOST option.**

**Explanation**
The HOST option was specified without indicating host name.

**System action**
The command is terminated.

**Operator response**
Reissue a valid host name at the HOST option. For a list of valid host names, contact the system operator.

**System programmer response**
None.

**Module**
LPR, LPQ, LPRM

**Procedure name**
ProcessOptions

**EZB0903E  The option option was not recognized.**

**Explanation**
The program cannot recognize the option you have entered. Valid options will be provided. This message will precede the valid options for your command.

**System action**
The command is terminated.

**Operator response**
Reissue the valid option using the LPR command.
Module
LPQ, LPR, LPRM, LPRSET

Procedure name
ProcessOptions

EZB0904I Use the ALL, HOST, PRINTER, TRACE, TYPE or VERSION options as needed.

Explanation
This message displays valid options that may be used with LPT, LPQ, and LPRM commands.

System action
The command is terminated.

Operator response
None.

System programmer response
None.

Module
LPQ, LPRM

Procedure name
ProcessOptions

EZB0905E Use a printer name after PRINTER option.

Explanation
The PRINTER option was specified without specifying a printer name.

System action
The command is terminated.

Operator response
Reissue the LPR command and the PRINTER option using a valid printer. For more information about the LPR command and the PRINTER option, see the z/OS Communications Server: IP User's Guide and Commands.

System programmer response
None.

Module
LPR, LPQ, LPRM
**Procedure name**
ProcessOptions

**EZB0906E**  Program error: Invalid option *option*.

**Explanation**
You specified an invalid option. To see a list of options, type HELP followed by your command or see z/OS Communications Server: IP User’s Guide and Commands for more information.

**System action**
The command is terminated.

**Operator response**
Reissue the LPR command and a valid option.

**System programmer response**
None.

**Module**
LPR, LPQ, LPRM, LPRSET

**Procedure name**
ProcessOptions

**EZB0907E**  Cannot get a printer name.

**Explanation**
The program cannot find a default printer name in the *user_id*.LASTING.GLOBALV data set.

**System action**
Processing continues.

**Operator response**
Use the LPRSET command to set up a default printer name. For more information see z/OS Communications Server: IP User’s Guide and Commands.

**System programmer response**
None.

**Module**
LPQ, LPR, LPRM

**Procedure name**
ProcessOptions

**EZB0908I**  Printer name from global variable PRINTER = “*printer*”
Explanation
LPR displays the default printer name, taken from the user_id.LASTING.GLOBALV data set. For more information about the user_id.LASTING.GLOBALV data set, see z/OS Communications Server: IP Configuration Reference.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR, LPQ, LPRM

Procedure name
ProcessOptions

EZB0909E Use the PRINTER option this way: PRINTER nameofprinter<@printerhost>

Explanation
You issued the PRINTER option with the at-sign without specifying the printer name.

System action
The command is terminated.

Operator response
Reissue the printer option using the correct syntax as described in the message.

System programmer response
None.

Module
LPR, LPRM, LPQ

Procedure name
ProcessOptions

EZB0910E Cannot get the printer host.

Explanation
The program cannot find the default host name from the user_id.LASTING.GLOBALV data set.

System action
The command is terminated.
Operator response
Specify the host name on the command line, or use the LPRSET command to set up a default host name. For more information about the LPRSET command see z/OS Communications Server: IP User’s Guide and Commands.

System programmer response
None.

Module
LPQ, LPR, LPRM

Procedure name
ProcessOptions

EZB0911I Host name from global variable PRTHOST = “host”

Explanation
If the LPR, LPQ, or LPRM command lines do not have a specified printer host, the GLOBALV variables PRINTER and PRTHOST are reviewed in the user_id.LASTING.GLOBALV data set and the name of the host in use is displayed.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR, LPQ, LPRM

Procedure name
ProcessOptions.

EZB0912E The printer name is not known.

Explanation
The program does not recognize the printer name.

System action
Processing continues.

Operator response
Check the printer name at the server machine, and reissue the command with a valid printer name or use the LPRSET command to set a default printer name at the specified host. See z/OS Communications Server: IP User’s Guide and Commands for more information about setting up a default printer in the user_id.LASTING.GLOBALV data set.
System programmer response
None.

Module
LPR, LPQ, LPRM

Procedure name
ProcessOptions

EZB0913E  The host name is not known.

Explanation
The program cannot determine the host name.

System action
The command is terminated.

Operator response
Use a valid host name with the command. The LPRSET command can be used to set up a default host name for a default printer. See z/OS Communications Server: IP User’s Guide and Commands for more information about setting up a default host in the user_id.LASTING.GLOBALV data set.

System programmer response
None.

Module
LPR, LPQ, LPRM

Procedure name
ProcessOptions

EZB0914E  Please specify the printer name and host either as a command option or with the LPRSET command.

Explanation
The program cannot determine the printer name or the printer host.

System action
Processing continues.

Operator response
Reissue your command with a valid printer name and host. The LPRSET command can be used to set a default printer and host using the GLOBALV variables in the user_id.LASTING.GLOBALV data set. For more information, see z/OS Communications Server: IP User’s Guide and Commands.

System programmer response
If the problem continues, obtain more information using the TYPE or TRACE functions.
Module
LPR, LPQ, LPRM

Procedure name
ProcessOptions

EZB0915I  Begin “cmd” to printer “printer” at host “foreignhost”

Explanation
This message indicates which printer and remote host are being used for your task. The task is indicated by “cmd” in the message, which can be LPR, LPQ, or LPRM.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR, LPQ, LPRM

Procedure name
ProcessOptions

EZB0916I  Sending command command argument: operand

Explanation
This message indicates the command was successfully sent to the remote host.

System action
The system continues processing.

Operator response
None.

System programmer response
None.

Module
LPR, LPQ, LPRM

Procedure name
SendCommand

EZB0917I  Command successfully sent
**Explanation**
The command was successfully sent and acknowledged by the remote print server.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPR, LPQ, LPRM

**Procedure name**
SendCommand

---

**EZB0918E Command was not sent successfully**

**Explanation**
The command was not acknowledged by the remote host; therefore, the transmission was unsuccessful.

**System action**
The command is terminated.

**Operator response**
Reenter the command using the correct syntax.

**System programmer response**
Assist the operator if necessary.

**Module**
LPQ, LPRM

**Procedure name**
SendCommand

---

**EZB0919I InitEmulation failed**

**Explanation**
An error occurred initializing EC mode emulation. EC mode emulation is required to run TCPIP on a non-EC mode machine.

**System action**
Processing terminated.
Operator response
Reinitialize EC mode to on. If a problem occurs contact the system programmer.

System programmer response
EC-mode emulation can only be accessed when the processor is running. If a program is running in a 24-bit addressing mode, the program can be changed to a 31-bit addressing mode, which causes a branch to a module residing above the 16Mb virtual storage. For more information see IBM Assembler Language Programming Book.

Module
LPQ, LPR, LPRM

Procedure name
LPR Main

EZB0920I Requesting TCP/IP service at date time

Explanation
This message indicates the time and date that LPR, LPQ, or LPRM requested TCPIP service. This message is displayed if the TRACE option is specified.

System action
Process continues.

Operator response
None.

System programmer response
None.

Module
LPQ, LPR, LPRM

Procedure name
LPR Main

EZB0921I Granted TCP/IP service at date time

Explanation
TCPIP service is provided for the command requested.

System action
Processing continues.

Operator response
None.

System programmer response
None.
EZB0922I  Resolving foreignhost at date time

Explanation
The remote host is being resolved at the indicated time and date.

System action
Processing continues.

Operator response
None.

System programmer response
None.

EZB0923I  Both the printer and the host name are not known.

Explanation
The program could not determine the printer name or the host name.

System action
The command is terminated.

Operator response
Check the host name and printer name for the correct information and reenter the command. Use the LPRSET command to set a default printer and host name. For more information see z/OS Communications Server: IP User's Guide and Commands.

System programmer response
None.

EZB0924I  Host host name resolved to host_addr at date time
**Explanation**  
The remote host name has been resolved to the indicated Internet address along with the time and date.

**System action**  
Processing continues.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
LPR, LPQ, LPRM

**Procedure name**  
LPR Main

**EZB0925I**  
TCP/IP turned on.

**Explanation**  
TCPIP services are now ready for use.

**System action**  
Processing continues with the next request.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
LPR, LPQ, LPRM

**Procedure name**  
LPR Main

**EZB0926I**  
Host “host” Domain “domain” TCP/IP Service Machine name

**Explanation**  
This message indicates the host name, its domain name equivalent, and the TCPIP service machine in use.

**System action**  
Processing continues.
**Module**
LPR, LPQ, LPRM

**Procedure name**
LPR Main

---

**EZB0927I** Trying to open with local port *port* to foreign host address *address*

**Explanation**
This message is issued while tracing is on and LPR is first initiated. TCPIP is attempting to open a connection with the stated local port to the stated foreign host address.

**System action**
Processing continues.

---

**EZB0928I** Connection open from local port *port* to foreign host address *address*

**Explanation**
This message is issued while tracing is on. TCPIP indicates that the specified local port has successfully connected to the specified foreign host address.

**System action**
Processing continues.

---

**Operator response**
None.

**System programmer response**
None.
Module
LPR, LPQ, LPRM

Procedure name
LPR Main

EZB0929I  Connected to host

Explanation
The local port has completed a successful telecommunication line to the remote port. This means your command has been accepted and transmitted through TCPIP services.

System action
Processing continues with the next request.

Operator response
None.

System programmer response
None.

Module
LPR, LPQ, LPRM

Procedure name
LPR Main

EZB0930I  Connection closed

Explanation
TCPIP services ends the connection when your task has completed processing.

System action
Processing ends successfully.

Operator response
None.

System programmer response
None.

Module
LPR, LPQ, LPRM

Procedure name
LPR Main, LPQ Main, or LPRM Main

EZB0931I  Notification: SayNotEn
**Explanation**
This message provides information about whether the data has been delivered and whether the connection state has been changed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPQ, LPRM

**Procedure name**
DrainConnection, ReceiveData

**EZB0932I**  
**NewState: SayConSt**

**Explanation**
The state of the connection to TCPIP has changed. The new state is indicated in the message.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPQ, LPRM

**Procedure name**
DrainConnection, ReceiveData

**EZB0933I**  
**ConnState: SayConSt**

**Explanation**
This message indicates the state of the connection to TCPIP.

**System action**
Processing continues.
Operator response
None.

System programmer response
None.

Module
LPQ, LPRM

Procedure name
DrainConnection, ReceiveData

EZB0934I  BytesToRead: bytes

Explanation
This message indicates the number of bytes contained in a transmission from TCPIP.

System action
The system continues processing.

Operator response
If the data is not acknowledged, TCPIP will retransmit the data to make sure that the data is received.

System programmer response
None.

Module
LPQ, LPRM

Procedure name
DrainConnection, ReceiveData

EZB0935I  BytesDelivered: bytes

Explanation
This message indicates the number of bytes delivered for your task, if the trace option is used.

System action
The system continues processing.

Operator response
None.

System programmer response
None.
Module
LPQ, LPRM

Procedure name
ReceiveBytes, ReceiveData

EJB0936E  BeginTcpip: errmsg (msgnum)

Explanation
The function BeginTcpip, which is used to start TCPIP service for LPQ or LPRM, was unsuccessful.
errmsg is the text of the message that describes the error.
msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmgsnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
Processing halts.

Operator response
Notify the system programmer of the error.

System programmer response
Respond as indicated by the message EZAmgsnum.

Module
LPQ, LPRM

Procedure name
LPR Main, LPQ Main or LPRM Main

EJB0939E  Could not get identity! (errmsg (msgnum))

Explanation
TCPIP was unable to identify the user requesting service.
errmsg is the text of the message that describes the error.
msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmgsnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
Processing ends.

Operator response
Respond as indicated by the message EZAmgsnum.

System programmer response
Assist the user as necessary.
Module
LPR, LPQ, LPRM

Procedure name
LPR Main, LPQ Main or LPRM Main

EZB0940E  Unknown host host

Explanation
The specified host does not exist or is entered incorrectly.

System action
Processing halts.

Operator response
Reenter the correct host name or address using the LPRSET command.

System programmer response
None.

Module
LPR, LPRM, LPQ

Procedure name
LPR Main

EZB0941E  Handle: errmsg (msgnum)

Explanation
The Handle procedure, which specifies what notifications to receive in a given set, was unsuccessful.
errmsg is the text of the message that describes the error.
msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
Processing halts.

Operator response
Notify the system programmer of the error.

System programmer response
Respond as indicated by the message EZAmsgnum.

Module
LPR, LPQ, LPRM
**EJB0942E** Connection to host failed.

**Explanation**
The attempt to connect to the specified host was not successful.

**System action**
Processing halts.

**Operator response**
Try to restart the connection by issuing the LPD command. For more information about the LPD command check the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
None.

**Module**
LPR, LPQ, LPRM

---

**EJB0943I** No local printer ports available now. Bind Conn failed.

**Explanation**
All of the local printer ports are either busy or not ready.

**System action**
Processing halts.

**Operator response**
Wait and try the command again.

**System programmer response**
None.

**Module**
LPR, LPQ, LPRM

---

**EJB0944E** Could not set option (errmsg (msgnum))

**Explanation**
The procedure that sets an option for a TCP connection was unsuccessful.
errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

**System action**
Processing halts.

**Operator response**
Notify the system programmer.

**System programmer response**
Respond as indicated by the message EZAmsgnum.

**Module**
LPQ, LPRM LPR, LPQ, LPRM

**Procedure name**
LPR Main, LPR Main, LPRM Main

---

**EZB0945E**  Could not send command *errmsg (msgnum)*

**Explanation**
The function which sends the commands was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

**System action**
Processing halts.

**Operator response**
Notify the system programmer.

**System programmer response**
Respond as indicated by the message EZAmsgnum.

**Module**
LPRM, LPQ

**Procedure name**
LPR Main

---

**EZB0946E**  Failed to read buffer *errmsg (msgnum)*

**Explanation**
The procedure that attempts to receive the data was unsuccessful.
errmsg is the text of the message that describes the error.

msgnum is the 4–digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

**System action**
Processing halts.

**Operator response**
Notify the system programmer.

**System programmer response**
Respond as indicated by the message EZAmgnum.

**Module**
LPRM, LPQ

**Procedure name**
LPR Main

**EZB0947E**  
TcpClose&colon errmsg (msgnum)

**Explanation**
The function that closes the connection was unsuccessful.
errmsg is the text of the message that describes the error.

msgnum is the 4–digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

**System action**
Processing halts.

**Operator response**
Notify the system programmer.

**System programmer response**
Respond as indicated by the message EZAmgnum.

**Module**
LPR, LPQ, LPRM

**Procedure name**
LPR Main

**EZB0948E**  
Could not abort connection errmsg (msgnum)

**Explanation**
The application was unable to abort a connection.
**errmsg** is the text of the message that describes the error.

**msgnum** is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in **errmsg**. For more information about this message, see message **EZAmgnum** in the z/OS Communications Server: IP Messages Volume 1 (EZA).

**System action**
Processing halts.

**Operator response**
Notify the system programmer.

**System programmer response**
Respond as indicated by the message **EZAmgnum**.

**Module**
LPR, LPQ, LPRM

**Procedure name**
LPR Main

**EZB0949E** Failed to send from SendACK. return code = \( rc \) and Error Number = \( errno \).

**Explanation**
The SendACK function, used by the server to relay acknowledgment, was not received by the host. The current port number and the user IP address are displayed with this message.

\( errno \) is the UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

**System action**
Processing continues.

**Operator response**
See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the \( errno \).

**System programmer response**
Assist the user as necessary.

**Module**
LPR

**Procedure name**
SendACK

**EZB0950E** Failed to send from SendCommand. return code = \( rc \) and Error Number = \( errno \).
Explanation
The function that sends the acknowledgment was unsuccessful. The command was not sent. The current port number and the user IP address are displayed with this message.

errno is the z/OS UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

System action
LPR halts.

Operator response
See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the errno.

System programmer response
Assist the user as necessary.

Module
LPR

Procedure name
SendCommand

EZB0951E Did not receive ACK for receive control file command

Explanation
SendCommand did not receive an acknowledgment code from the receiving host, and the control file command was not received.

System action
LPR halts.

Operator response
Try the command again making sure the host is ready and the connection is open.

System programmer response
Assist the user as necessary.

Module
LPR

Procedure name
SendControlFile

EZB0952E Failed to send control line block. return code = rc and Error Number = errno.

Explanation
The function that sends the data was unsuccessful. The current port number and the user IP address are displayed with this message.
errno is the z/OS UNIX System Services return code. Return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

**System action**
LPR halts.

**Operator response**
See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the errno.

**System programmer response**
Assist the user as necessary.

**Module**
LPR

**Procedure name**
SendControlFile

**EZB0953E** Did not receive ACK for control file

**Explanation**
The function TcpFReceive returned an FRECEIVEerror, indicating that the receive request was rejected. The control file was not received.

**System action**
LPR halts.

**Operator response**
Make sure the receiving host is ready and try the request again.

**System programmer response**
Assist the user as necessary.

**Module**
LPR

**Procedure name**
SendControlFile

**EZB0954E** File could not be opened.

**Explanation**
LPR was unable to open a data set.

**System action**
Processing continues.
Operator response
Verify that the data set is in storage accessible to LPR.

System programmer response
None.

Module
LPR

Procedure name
SendControlFile

EZB0955E  File contains no valid print lines.

Explanation
The data set is empty.

System action
Processing continues.

Operator response
Make sure the data set you wish to print contains at least one character.

System programmer response
None.

Module
LPR

Procedure name
SendControlFile

EZB0956E  Failed to send data block errmsg (msgnum)

Explanation
The attempt to send the information was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4–digit numeric portion of the message identifier of the EZA message whose text is displayed in errmsg. For more information about this message, see message EZAmsgnum in the z/OS Communications Server: IP Messages Volume 1 (EZA).

System action
Processing continues.

Operator response
Notify the system programmer.
System programmer response
Respond as indicated by the message EZAmsgnum.

Module
LPR

Procedure name
SendLineFlush

EZB0957E Did not receive ACK for receive data file command. Code = code.

Explanation
The host did not receive an acknowledgment from the LPR receive data command.

System action
LPR halts.

Operator response
See z/OS Communications Server: IP and SNA Codes for information about return codes and error numbers. The current port number and the user IP address are displayed with this message.

System programmer response
Assist the user as necessary.

Module
LPR

Procedure name
SendDataFile

EZB0958E File could not be opened.

Explanation
The specified data set could not be opened.

System action
LPR halts.

Operator response
Verify that you have access to the data set and try to open it again.

System programmer response
Assist the user as necessary.
**Procedure name**
SendDataFile

**EZB0959E  Did not receive ACK for data file**

**Explanation**
The remote host did not acknowledge that it received a data set.

**System action**
If no acknowledgement is received, the data set is retransmitted. Processing continues.

**Operator response**
Notify the system programmer of the error.

**System programmer response**
Check why LPD did not send acknowledgement. Starting LPD (server) with “DEBUG” on, will help.

**Module**
LPR

**Procedure name**
SendDataFile

**EZB0960E  Did not receive ACK after close**

**Explanation**
The remote host did not send a message acknowledging that it has closed. The connection remains open.

**System action**
Processing continues.

**Operator response**
Try to close the connection again.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
DrainConnection

**EZB0961I  Control file name is dataset name**

**Explanation**
The name of the control data set is displayed.
**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
MakeFileNames

---

**EZB0962I  Data file name is dataset name**

**Explanation**
The name of the current data set is displayed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
MakeFileNames

---

**EZB0965E  Use the form: command name DataSetName.**

**Explanation**
This message displays the correct format for the indicated command. For more information about the commands, see the z/OS Communications Server: IP User's Guide and Commands.

**System action**
The command is not processed.

**Operator response**
Reenter the command using the form given in the z/OS Communications Server: IP User's Guide and Commands.
System programmer response
Assist the user as necessary.

Module
LPR

Procedure name
ProcessOperands

EZB0967E  Data Set name “name” invalid.

Explanation
The specified data set name is incorrect.

System action
The command is not processed.

Operator response
Make sure that the data set name uses the correct format and syntax and that the data set is in storage accessible to the host.

System programmer response
Assist the user as necessary.

Module
LPD, LPR

Procedure name
ProcessOperands

EZB0968E  Data Set “name” not found or inaccessible, return code = rc.

Explanation
This message may issue one of the following return codes:

Return code  Description
4      Data set name not found
12     Data set name is missing
24     Data set is an unprintable VSAM file

If another return code is issued, this indicates that the data set attributes may make the data set inaccessible.

System action
LPR halts.
Operator response
Reenter the LPR command using the correct data set name. If the data set name is correct, check the data set attributes.

System programmer response
Assist the user as necessary.

Module
LPR

Procedure name
ProcessOperands

EZB0969E Data Set “name” does not contain member “member”.

Explanation
The specified data set name does not contain the specified member name.

System action
The LPR command is not processed.

Operator response
Reenter the command using the correct data set name and the correct member name.

System programmer response
None.

Module
LPR

Procedure name
ProcessOperands

EZB0970E Data Set “name” invalid organization.

Explanation
The function FindDSName exited abnormally because the file's organization was incorrect.

System action
Processing continues.

Operator response
Recheck the organization and reenter the file. Notify the system programmer if the problem persists.

System programmer response
Assist the user as necessary.
Module
LPR

Procedure name
ProcessOperands

EZB0971E  Use a matching quotation mark at the end of the string.

Explanation
There is a missing quotation mark at the end of the string.

System action
The LPR command is not processed.

Operator response
Make sure there are appropriate quotation marks and reenter the string.

System programmer response
None.

Module
LPR

Procedure name
LPRToken

EZB0973E  Use either BINARY or NOBINARY but not both.

Explanation
Either BINARY or NOBINARY can be chosen. Data cannot be sent as BINARY and NOBINARY.

System action
The LPR command is not processed.

Operator response
Reenter the LPR command specifying either the BINARY or NOBINARY options.

System programmer response
None.

Module
LPR

Procedure name
ProcessOptions

EZB0974E  Use a token after the CLASS option.
Explanation
After the CLASS option, you must enter a name of a host. If you do not want to name another host name, the CLASS option is not required. For more information about the LPR command and its options, see the z/OS Communications Server: IP User’s Guide and Commands.

System action
LPR command is not processed.

Operator response
Reenter the LPR command and the CLASS option using a name of a host.

System programmer response
Assist the user as necessary.

Module
LPR

Procedure name
ProcessOptions

EZB0975E Use a count after the COPIES option.

Explanation
This message indicates that there is no number following the COPIES option. The COPIES option specifies how many copies are to be printed. If this option is not used, one copy will be printed.

System action
LPR command is not processed.

Operator response
Reenter the LPR command with the COPIES option and fill in a number, specifying the number of copies you want printed. If only one copy is needed, the COPIES option is not required.

System programmer response
None.

Module
LPR

Procedure name
ProcessOptions

EZB0976E Use a letter after the FILTER option.

Explanation
No filter was specified after the filter option in the LPR command. The filter option specifies the type of processing to be done with the data.
System action
The LPR command is not processed.

Operator response
Reenter the LPR command using the filter option with a valid filter. See z/OS Communications Server: IP User's Guide and Commands for more information about the LPR command.

System programmer response
Assist the user as necessary.

Module
LPR

Procedure name
ProcessOptions

EZB0977E Use a number after the INDENT option.

Explanation
A numeric argument was not specified for the INDENT option of the LPR command.

System action
LPR command is not processed.

Operator response
Reenter the LPR command, specifying a numeric argument for the INDENT option. For more information about the LPR command and the INDENT option, see the z/OS Communications Server: IP User's Guide and Commands.

System programmer response
None.

Module
LPR

Procedure name
ProcessOptions

EZB0978E Use a token after the JOB option.

Explanation
You must specify a name when using the JOB option. The name is the printing job's description to the remote system.

System action
The LPR command is not processed.
Operator response
Reenter the LPR command with a name following the JOB option. For more information about the JOB option of the LPR command see the z/OS Communications Server: IP User's Guide and Commands.

System programmer response
None.

Module
LPR

Procedure name
ProcessOptions

EZB0979E Use a number after the LINECOUNT option.

Explanation
You must enter a number following the LINECOUNT option to specify the number of lines to be printed before a new heading is printed.

System action
The LPR command is not processed.

Operator response
Reenter the LPR command with the LINECOUNT option using a valid number. For more information see the z/OS Communications Server: IP User's Guide and Commands.

System programmer response
None.

Module
LPR

Procedure name
ProcessOptions

EZB0980E Use an identification after NAME option.

Explanation
You did not specify a name after the NAME option in the LPR command. (The NAME option specifies the job information to be provided by the remote system in response to a query.)

System action
The LPR command is not processed.

Operator response
Resubmit the LPR command supplying a name after the NAME option. For more information about the LPR command and the NAME option, see the z/OS Communications Server: IP User's Guide and Commands.
**System programmer response**
None.

**Module**
LPR

**Procedure name**
ProcessOptions

---

**EZB0981E**  Use a title after the TITLE option.

**Explanation**
You did not specify the title after the TITLE option in the LPR command.

**System action**
The LPR command is not processed.

**Operator response**
Resubmit the LPR command supplying a name after the TITLE option. For more information about the LPR command and the TITLE option, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
None.

---

**Module**
LPR

**Procedure name**
ProcessOptions

---

**EZB0982E**  Use a number after the TOPMARGIN option.

**Explanation**
You did not specify a number after the TOPMARGIN option in the LPR command. (This number specifies the number of lines designated for the top margin).

**System action**
The LPR command is not processed.

**Operator response**
Resubmit the LPR command supplying the number of lines to be designated for the top margin. For more information about the LPR command and the TITLE option, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
None.
EZB0983E  Use a title after the WIDTH option.

Explanation
You did not specify the line width after the WIDTH option in the LPR command.

System action
The LPR command is not processed.

Operator response
Resubmit the LPR command supplying the line width of the data set. For more information about the LPR command and the WIDTH option see the z/OS Communications Server: IP User’s Guide and Commands.

System programmer response
None.

EZB0984E  Use either the NOPOSTSCRIPT or POSTSCRIPT option but not both.

Explanation
You selected both the POSTSCRIPT and the NOPOSTSCRIPT options in the LPR command. You can select only one of these options.

System action
The LPR command is not processed.

Operator response
Reissue the command with the correct option. For more information about the LPR command and the POSTSCRIPT or NOPOSTSCRIPT options, see the z/OS Communications Server: IP User’s Guide and Commands.

System programmer response
None.
Use either the POSTSCRIPT or LANDSCAPE option but not both.

**Explanation**
You selected both the POSTSCRIPT and LANDSCAPE options in the LPR command. You can select only one of these options.

**System action**
The LPR command is not processed.

**Operator response**
Reissue the command with the correct option. For more information about the LPR command and the POSTSCRIPT or NOPOSTSCRIPT options, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
ProcessOptions

Use either the BINARY OR LANDSCAPE option but not both.

**Explanation**
You selected both the BINARY and LANDSCAPE options in the LPR command. You may select only one of these options. If you select the LANDSCAPE option, the only valid filter option that you can specify explicitly is the "o" filter.

**System action**
The LPR command is not processed.

**Operator response**
Reissue the command using the correct option. For more information about the LPR command and the BINARY or LANDSCAPE options, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
ProcessOptions

Use either the FILTER or LANDSCAPE option but not both.
**Explanation**
You selected both the FILTER and LANDSCAPE options in the LPR command. You may select only one of these options. If you select the LANDSCAPE option, the only valid filter option that you can specify explicitly is the "o" filter.

**System action**
The LPR command is not processed.

**Operator response**
Reissue the command with the correct option. For more information about the LPR command and the FILTER or LANDSCAPE options, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
ProcessOptions

**EZB0988I**  PostScript program is number bytes.

**Explanation**
This message specifies the number of bytes in the PostScript program.

**System action**
The LPR command is not processed.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
ProcessOptions

**EZB0989I**  Use these options.
**EZB0990I**  additional options
**EZB0991I**  additional options or last options

**Explanation**
TCPIP issues this message when one or more incorrect options is specified in the LPR command. It provides a list of valid options.
System action
The LPR command is not processed.

Operator response
Reissue the LPR command using valid options.

System programmer response
None.

Module
LPR, LPRSET

Procedure name
ProcessOptions

EZB0992I File contains PostScript.

Explanation
TCPIP issues this message when tracing is on and PostScript characters are detected in the data set.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
SendControlFile

EZB0993E Use the program which produced the PostScript to change to landscape display.

Explanation
The LANDSCAPE option is specified in the LPR command and the file to be printed is a PostScript data set.

System action
The LPR command is not processed.

Operator response
Redefine the data set for landscape display using the original application program that created the POSTSCRIPT file. Then reissue the LPR command without the LANDSCAPE option. See also EZB1010E.
EZB0994E Use the NOPOSTSCRIPT option to prevent special processing of PostScript files.

Explanation
The LPR command contains both POSTSCRIPT and CC options.

System action
The LPR command is not processed.

Operator response
Reissue the LPR command using the NOPOSTSCRIPT option in place of the POSTSCRIPT option. For more information about the LPR command and the POSTSCRIPT and NOPOSTSCRIPT options, see the z/OS Communications Server: IP User’s Guide and Commands.

EZB0995W Ignoring “buffer”

Explanation
TCP/IP issues this message when an incorrect carriage control character is detected. The lines containing those incorrect characters are ignored.

System action
Processing continues.

Operator response
Correct the carriage control characters in the data set.

System programmer response
None.
**Procedure name**
SendControlFile

**EZB0996W**  number lines with invalid carriage control characters deleted.

**Explanation**
TCPIP issues this message when incorrect carriage control characters are detected in the data set that is being processed. The lines containing the incorrect carriage control characters are deleted.

**System action**
Processing continues.

**Operator response**
Correct the carriage control characters in the data set and resubmit the job.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
SendControlFile

**EZB0997I**  Byte size check starts at date time

**Explanation**
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The date and time are recorded before performing a byte size check of the data set.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
SendDataFile

**EZB0998I**  Byte size check ends at date time
Explanation
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The date and time are recorded after performing a byte size check of the data set.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
SendDataFile

EZB0999I Send command starts at date time

Explanation
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. Arguments are passed prior to the actual data transfer. The date and time are recorded before those arguments are passed.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
SendDataFile
Chapter 3. EZB1xxx messages

**EZB1000I**  Send command ends at *date time*

*Explanation*
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. After arguments are passed and acknowledgments are received, the date and time are recorded.

*System action*
Processing continues.

*Operator response*
None.

*System programmer response*
None.

*Module*
LPR

*Procedure name*
SendDataFile

**EZB1001I**  Send data starts at *date time*

*Explanation*
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. Before the data set is sent to the remote printer, the date and time are recorded.

*System action*
Processing continues.

*Operator response*
None.

*System programmer response*
None.

*Module*
LPR

*Procedure name*
SendDataFile

**EZB1002I**  Send data ends at *date time*
Explanation
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. When the data set is received by the remote printer, the date and time are recorded.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
SendDataFile

**EZB1003I** Send ACK starts at date time

Explanation
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. TCPIP initiates an acknowledgment process. The date and time are recorded.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
SendDataFile

**EZB1004I** Send ACK ends at date time

Explanation
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. When the requested acknowledgments are received, the acknowledgment process is complete; the date and time are recorded.
System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
SendDataFile

EJB1005I  Draining the connection.

Explanation
This message occurs while the data is sent from the host to the remote printer and tracing is on. Before the connection between the host and the remote printer closes, the information is taken in and processed by the printer.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
DrainConnection

EJB1006E  Host host did not accept printer name printer

Explanation
The indicated host did not recognize the indicated printer.

System action
The LPR command is not processed.

Operator response
Resubmit the job using the correct host printer name.
System programmer response
None.

Module
LPR

Procedure name
DrainConnection

EZB1007I  Connection still receiving - aborting it.

Explanation
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The remote host continues to receive information. When all the information is received, the connection is aborted.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
DrainConnection

EZB1008I  Connection aborted.

Explanation
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. TCPIP ended the connection with the remote host.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR
**Procedure name**  
DrainConnection

**EZB1009I**  
Data file sent.

**Explanation**  
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The data set has been successfully sent.

**System action**  
Processing continues.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
LPR

**Procedure name**  
SendDataFile

**EZB1010E**  
Use the program which produced the PostScript to control pagination.

**Explanation**  
This message follows EZB0993E and is issued when the LANDSCAPE option is specified in the LPR command and the file to be printed is a PostScript data set.

**System action**  
The LPR command is not processed.

**Operator response**  
Redefine the data set to control pagination using the original application program that created the POSTSCRIPT file. Then reissue the LPR command without the LANDSCAPE option.

**System programmer response**  
None.

**Module**  
LPR

**Procedure name**  
SendControlFile

**EZB1011I**  
Queuing control line “number”
Explanation
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The commands are retrieved in the order in which they were submitted.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
QueueControlLine

EJB1012I Receiving ACK

Explanation
TCPIP issues this message when tracing is on and a data set is sent to a remote printer using the LPR command. TCPIP is receiving acknowledgments from the remote host.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPR

Procedure name
ReceiveByte

EJB1013I ReceiveACK: word for byte value rc

Explanation
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. TCPIP is receiving acknowledgments from the remote host.

System action
Processing continues.
**Operator response**
None.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
ReceiveByte

**EZB1014I Sending ACK**

**Explanation**
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The remote host is sending acknowledgments to TCPIP.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
SendACK

**EZB1015I ACK successfully sent**

**Explanation**
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The acknowledgment process has been successfully completed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
**Module**
LPR

**Procedure name**
SendACK

**EZB1016E**  Did not receive an ACK for command. Code=rc

**Explanation**
TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. TCPIP did not receive an acknowledgment from the remote host. The return code is displayed.

**System action**
The LPR command is not processed.

**Operator response**
Resubmit the LPR command.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
SendCommand

**EZB1017I**  Control data sent

**Explanation**
A control data set was sent to the remote host. The control data set contains the printer setup information.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
SendControlFile

**EZB1018I**  Control file sent
**Explanation**
A control data set was sent to the remote host. The control data set contains the information to be printed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
SendControlFile

---

**EZB1019E**  
Use the form: LPRSET nameofPrinter@printerhost.

**Explanation**
You specified an incorrect format for the LPRSET command.

**System action**
The LPRSET command is not processed.

**Operator response**
Reissue the LPRSET command with the correct syntax (nameofPrinter@printerhost). For more information about the LPRSET command, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
None.

**Module**
LPR, LPRSET

**Procedure name**
ProcessOperands

---

**EZB1020I**  
Your LPR printer is currently set to printer at host

**Explanation**
This message is displayed when the LPRSET (QUERY command is issued. The LPR printer name and host are identified.

**System action**
Processing continues.
Operator response
None.

System programmer response
None.

Module
LPRSET

Procedure name
ProcessQueryOption

**EZB1021I**  Append of host name to LASTING.GLOBALV status rc

Explanation
This message is displayed when the LPRSET command is issued while tracing is on. The remote host name is added to the LASTING.GLOBALV data set. This message is followed by EZB1022I and EZB1023I.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
LPRSET

Procedure name
ProcessArguments

**EZB1022I**  Append of printer name to LASTING.GLOBALV status rc

Explanation
This message is displayed when the LPRSET command is issued while tracing is on. The printer name is added to the LASTING.GLOBALV data set. This message appears with EZB1021I and EZB1023I.

System action
Processing continues.

Operator response
None.

System programmer response
None.
**Module**
LPRSET

**Procedure name**
ProcessArguments

**EZB1023I** Printer set to *printer at host*

**Explanation**
This message is displayed when the LPRSET command is issued while tracing is on. The LPR printer name and host are identified. This message is preceded by EZB1021I and EZB1022I.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

---

**Module**
LPRSET

**Procedure name**
ProcessArguments

**EZB1024E** System error *error* setting printer.

**Explanation**
This message occurs as a result of a failure while attempting to set the LPR printer. For more information about the LPRSET command, see the z/OS Communications Server: IP User's Guide and Commands.

**System action**
The LPRSET command is not processed.

**Operator response**
Check the remote printer and resubmit the job.

**System programmer response**
None.

---

**Module**
LPRSET

**Procedure name**
ProcessArguments

**EZB1025E** Cannot use BINARY with a DBCS Translation Mode.
The BINARY parameter may not be specified with any other of the following LPR command line parameters:

- JIS78KJ
- JIS83KJ
- SJISKANJI
- EUCKANJI
- IBMKANJI
- HANGEUL
- KSC5601
- TCHINESE

The LPR command is not processed.

Consult the system programmer.

Specify either the BINARY parameter or a DBCS translation mode parameter. See z/OS Communications Server: IP User's Guide and Commands for more information about using DBCS conversion in LPR.

Module
LPR

Procedure name
ProcessDbcsOption

EZB1026E Cannot use multiple DBCS Translation Modes.

More than one of the following LPR command line parameters was specified:

- JIS78KJ
- JIS83KJ
- SJISKANJI
- EUCKANJI
- IBMKANJI
- HANGEUL
- KSC5601
- TCHINESE

The LPR command is not processed.

Consult the system programmer.
System programmer response
Specify only one of the above DBCS translation mode parameters. See z/OS Communications Server: IP User's Guide and Commands for more information about using DBCS conversion in LPR.

Module
LPR

Procedure name
ProcessDbcsOption

EZB1027E Unable to Load DBCS_translate_table_dataset

Explanation
LPR was attempting to load a DBCS translation table corresponding to the DBCS conversion mode parameter specified on the LPR command line. All data sets in the search order hierarchy for the required translate table data set, either do not exist, or do not contain data in the required format for DBCS binary translate tables.

System action
The LPR command is not processed.

Operator response
Consult the system programmer.

System programmer response
Configure a valid DBCS binary translate table data set in the search order hierarchy for the required DBCS translation table. See z/OS Communications Server: IP User's Guide and Commands for more information about the loading and customizing of DBCS translation tables.

Module
LPR

Procedure name
LoadDbcsTables


Explanation
This message is issued because the LPD server failed to send the PostScript ID that is contained in the PostScript data file.

errno is the UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

System action
LPR halts.

Operator response
Notify the system programmer.
System programmer response
See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the errno.

Module
LPR

Procedure name
SendDataFile

EZB1044E Specify a user name after USER option.

Explanation
The LPR command was entered with no user name specified for the USER option.

System action
LPR halts.

Operator response
Reenter the LPR command, specifying a user name after the USER option. For more information, see the z/OS Communications Server: IP User’s Guide and Commands.

System programmer response
None.

Module
LPR

Procedure name
ProcessOptions

EZB1045E Send ACK failed. return code = rc. Error Number = errno.

Explanation
The client machine failed while attempting to send acknowledgments to the server machine. The current port number and the user IP address are displayed with this message.

errno is the z/OS UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

System action
LPR halts.

Operator response
See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the errno.

System programmer response
None.
**Module**
LPR

**Procedure name**
SendDataFile

**EZB1046E**  
Shutdown failed. return code = rc. Error Number = errno.

**Explanation**
While attempting to close the TCP connections on both the client and server sides, an error was detected. The current port number and the user IP address are displayed with this message.

*errno* is the z/OS UNIX System Services return code. return codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

**System action**
LPR halts.

**Operator response**
See *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

**System programmer response**
None.

**Module**
LPR

**Procedure name**
SendDataFile

**EZB1047E**  
Send Control File failed. return code = rc. Error Number = errno.

**Explanation**
An error was detected while attempting to send the file format information. The current port number and the user IP address are displayed with this message.

*errno* is the z/OS UNIX System Services return code. return codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

**System action**
LPR halts.

**Operator response**
See *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

**System programmer response**
None.
Module
LPR

Procedure name
SendDataFile

**EZB1048E**  Send Data File failed. return code = *rc*. Error Number = *errno*.

**Explanation**
An error was detected while attempting to send the data file containing the file data. The current port number and the user IP address are displayed with this message.

*errno* is the z/OS UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

**System action**
LPR halts.

**Operator response**
See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the *errno*.

**System programmer response**
None.

Module
LPR

Procedure name
SendDataFile

**EZB1049E**  Send printer command did not receive ACK. ACK message = *message*.

**Explanation**
No acknowledgment was received from the server while attempting to send a printer command. An acknowledgment message was passed. The current port number and the user IP address are displayed with this message.

**System action**
LPR halts.

**Operator response**
Use the ACK message to determine and correct the problem.

**System programmer response**
Assist the user as necessary.
Procedure name
SendDataFile

EZB1050E Failed to Send Data. return code = rc. Error Number = errno port number = port remote ip address = ipaddr

Explanation
An error was detected while attempting to send data to the server machine.

rc is the return code from the subroutine that was called.

errno is the z/OS UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

port is the number of the connecting port.

ipaddr is the remote IP address.

System action
LPR halts.

Operator response
If rc is -1 and errno is 0, then there was a read error while trying to access the DBCS translation tables. Check your translation tables and their location. Correct any problems with the DBCS translation tables and try the LPR command again.

If rc is -2 and errno is 0, then there are DBCS characters that are not valid in the print file that you are trying to send. Turn on the trace option in LPR to see more messages from the translation routines. Correct the print file and try the LPR command again.

If rc is -1 and errno is nonzero, then look up the errno in the z/OS UNIX System Services Messages and Codes for help in diagnosing the exact problem.

System programmer response
None.

Module
LPR

Procedure name
SendDataFile

EZB1051E Failed to Open connection to Port Number = port. return code = rc error number = errno port number = port remote ip address = ipaddr

Explanation
The attempt to open a connection at the stated port was unsuccessful. The requested port, return code, error number, current port, and the IP address are displayed with this message.

errno is the z/OS UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

System action
LPR halts.
**Operator response**

Try to access the stated port number again. If the problem persists, notify the system programmer.

**System programmer response**

See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the **errno**.

**Module**

LPR

**Procedure name**

SendDataFile

**EZB1052E** StartTcpip: return code = rc and Error Number = **errno**.

**Explanation**

An error was detected while attempting to initialize TCPIP.

**errno** is the z/OS UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

**System action**

LPR halts.

**Operator response**

See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the **errno**.

**System programmer response**

Assist the user as necessary.

**Module**

LPR

**Procedure name**

SendDataFile

**EZB1053E** Could not set option. return code = rc and Error Number = **errno**.

**Explanation**

While attempting to set the socket KeepAlive option, an error was detected.

**errno** is the z/OS UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

**System action**

LPR halts.
Operator response
See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the errno.

System programmer response
None.

Module
LPR

Procedure name
SendDataFile

**EZB1054E**  TcpClose: return code = rc and Error Number = errno.

Explanation
While attempting to close the TCP connection on both the client and server machines an error was detected. The current port number and the user IP address are displayed with this message.

errno is the z/OS UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

System action
LPR halts.

Operator response
See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the errno.

System programmer response
None.

Module
LPR

Procedure name
SendDataFile

**EZB1055E**  Use a three-digit number after the JNUM option.

Explanation
The value specified for JNUM is not valid. It is either missing, greater than or less than three digits, or is not numeric. The JNUM value can be any 3-digit number in the range 000—999. The print job is not processed.

System action
LPR halts.

Operator response
Correct the JNUM value specified on the LPR command.
**System programmer response**
Assist the user as necessary.

**Module**
LPR

**Procedure name**
ProcessOptions

**EZB1056E  Invalid Internet address **Foreignhost**.**

**Explanation**
You provided an address containing numbers greater than 255. This is an invalid IP address for LPR.

**System action**
LPR halts.

**Operator response**
Check the IP address to which you are sending your output.

**System programmer response**
Assist the user as necessary.

**Module**
LPR

**Procedure name**
MainLoop

**EZB1057I  Loaded translation table from data set name**

**Explanation**
The fully qualified data set name used by LPR for the translation tables is displayed. These tables are used to translate EBCDIC to ASCII and ASCII to EBCDIC. This message may be issued more than once if more than one table is loaded. The last table loaded of a particular type will be the one used. (This will be the one cited in the last occurrence of this message).

**System action**
Processing continues.

**Operator response**
If the data set name displayed is not the one desired, verify that the expected data set exists. The search order used by LPR to find the translation data set is described in the *z/OS Communications Server: IP Configuration Reference*. If the data set name displayed is empty (""), then no translate data set was found in the search order. Create the desired translate table, or correct the TRANSLATE option specified on the LPR command.
System programmer response
Verify that the system translate tables are correctly installed under the correct high level qualifier (for example, hlq.STANDARD.TCPXLBIN). For information about determining the high level qualifier used, see the z/OS Communications Server: IP Configuration Reference.

Module
LPR

Procedure name
ProcessOptions

EJB1058E Data Set dataset_name not found

Explanation
The line printer requester (LPR) was unable to find the data set that was used in the invocation of the LPR command.

In the message text:

dataset_name
  The name of the data set that could not be found.

System action
LPR ends.

Operator response
No action is needed.

System programmer response
Assist the user, as necessary.

User response
Reenter the LPR command using a correct data set name. If the data set name is correct, verify that the data set exists and is cataloged. If the data set does exist and is cataloged, contact the system programmer.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: LPT

Module
LPR

Routing code
Not applicable.

Descriptor code
Not applicable.
EZB1058E  Data Set USER1.TCPIP.DATA2 not found

EZB1059E  Data Set dataset_name larger than 2,147,483,647 bytes

Explanation
The data set is too large to be sent using the LPR command.
In the message text:

*dataset_name*
  The name of the data set that is being sent.

System action
LPR ends.

Operator response
None.

System programmer response
Not Applicable.

User response
You can either make the data set smaller or use another mechanism to send the data set to a remote printer.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: LPT

Module
LPR

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZB1059E  Data Set USER1.TCPIP.DATA larger than 2,147,483,647 bytes

EZB1098I  return code = rc.  Error Number = errno.
**Explanation**

See the user response. The text of this message is usually appended to the end of other messages.

`errno` is the z/OS UNIX System Services return code. return codes are listed and described in the z/OS UNIX System Services Messages and Codes.

**System action**

Depending on the return code and error number in this message, LPR process may continue or may terminate.

**Operator response**

See z/OS Communications Server: IP and SNA Codes for information about TCP/IP return codes and the z/OS UNIX System Services Messages and Codes for information about the `errno`. If the return code is -1 and error number is 0, then negative acknowledgment was received from the server. Turn on the trace for the server to find out why the server rejected the LPR request.

**System programmer response**

None.

---

**Module**

LPR

**Procedure name**

SendDataFile

**EZB1099I**  Port Number = `port`. Remote IP Addr = `target address`

**Explanation**

See the user response. The text of this message is usually appended to the end of other messages. This message displays the port number and the IP address currently in use.

**System action**

Processing continues.

**Operator response**

None.

**System programmer response**

None.

---

**Module**

LPR

**Procedure name**

SendDataFile

**EZB1100I**  Cannot load translate table with name - `name` using defaults

---

EJB1xxx messages  167
Explanation
The Line Printer Request (LPR) attempted to load an SBCS translation table corresponding to the TRANSLATETABLE or XLATETABLE parameter specified on the LPR command line. All data sets in the search order hierarchy for the required translate table data set, either do not exist, or do not contain data in the required format for SBCS binary translate tables.

name is an input string that becomes part of the translation table data set name (for example, hlq.name.TCPXLBIN).

System action
LPR continues, however, default hardcoded tables are used for translations.

Operator response
Notify the system programmer.

System programmer response
Configure a valid SBCS binary translate table data set in the search order hierarchy for the required SBCS translation table. See z/OS Communications Server: IP Configuration Reference for more information about using translation tables, including search order hierarchy and customization.

Module
LPR

Procedure name
ProcessOptions

EZB1200E Error parsing argument “option” (specifier); unknown kind

Explanation
XWindows has encountered an option or specifier that it does not recognize in a user-submitted command line. The command line is not processed.

System action
XWindows continues.

Operator response
Correct the syntax and resubmit the command.

System programmer response
None.

Module
PARSECMD

Procedure name
_XReportParseError

EZB1201E Bad image image_type: error found in routine: routine
**Explanation**
Xwindows received an incorrect image type from the indicated routine. The image type and Xwindows error code are displayed in the message.

**System action**
Xwindows halts.

**Operator response**
Notify the system programmer of the error.

**System programmer response**
Use the error code given in the message and *z/OS Communications Server: IP and SNA Codes* to determine the cause of the error, and correct the indicated routine as necessary.

**Module**
XIMUTIL

**Procedure name**
_XReportBadImage

---

**EZB1202E**  Xlib: warning, client built for newer rev (*rev_number*) than server (*rev_number*)!

**Explanation**
The Xwindows client code was built for a newer version of Xwindows than the Xwindows server code. This can result in compatibility errors.

**System action**
Xwindows halts.

**Operator response**
Notify the system programmer of the error.

**System programmer response**
Correct either the client code or the server code to eliminate the revision disparity and rebuild the code.

**Module**
XOPENDIS

**Procedure name**
_XRead

---

**EZB1203E**  Xlib: warning, client is protocol rev *revision*, server is rev *revision*!

**Explanation**
The Xwindows server is using a different revision of the Xwindows code than the client. This can cause compatibility problems.
System action
Xwindows halts.

Operator response
Notify the system programmer of the error.

System programmer response
Correct the disparity in revision levels between the client and the server and restart Xwindows.

Module
XOPENDIS

Procedure name
_XRead

EZB1204E  Xlib: connection to server refused by server

Explanation
This message indicates that no authorization exists between the Xwindow server and the client program. The error causes the server to refuse the connection. The connection is not established.

System action
No Xwindows session can be opened for this client.

Operator response
Issue an Xhost command from the Xserver side.

System programmer response
None.

Module
XOPENDIS

Procedure name
Main

EZB1205E  Xlib: sequence lost (0xnumber_received > 0xnumber_expected) in reply type 0xreply_type

Explanation
The Xwindows server received a reply packet with a sequence number greater than the sequence number expected, indicating that a reply packet has been lost.

System action
Xwindows halts processing.

Operator response
Notify the system programmer of the error.
System programmer response
Check the indicated Xclient to determine why the packet was lost and respond as indicated.

Module
XLIBINT

Procedure name
_XSetLastRequestRead

EJB1206E  Xlib: unhandled wire event! event number = number, display = display_number

Explanation
Xwindows encountered an unknown event during conversion between the host format and the wire format.

System action
Xwindows halts.

Operator response
Notify the system programmer of the error.

System programmer response
Check the XEvent structure to make sure it is compatible with the current host. For more information, see the Xwindows documentation.

Module
XLIBINT

Procedure name
_XUnknownWireEvent

EJB1207E  Xlib: unhandled native event! event number = number, display = display_number

Explanation
Xwindows encountered an unknown event while reformatting a wire event to the host structure.

System action
Xwindows halts.

Operator response
Notify the system programmer of the error.

System programmer response
Check the XEvent structure to make sure that it is compatible with the current host. For more information see the Xwindows documentation.

Module
XLIBINT
**Procedure name**  
_XUnknownNativeEvent

**EZB1208E**  
**XIO: fatal IO error** *error_description* on X server *address_of_server*

**EZB1209E**  
after *number* requests (*number* known processed) with *number* events remaining. *err_no* (error) on X server “*server*”

**Explanation**  
The Xwindows server encountered an I/O error during processing. The type of error is indicated in the message. The error description portion of this message indicates the cause of the error.

**System action**  
Xwindows halts.

**Operator response**  
Notify the system programmer of the error.

**System programmer response**  
Respond as indicated by the *error_description* portion of this message.

**Module**  
XLIBINT

**Procedure name**  
_XDefaultIOError

**EZB1210I**  
The connection was probably broken by a server shutdown or KillClient.

**Explanation**  
If this message follows messages EZB1208E and EZB1209E, it indicates that the error displayed in those messages was probably caused by the shutdown of the Xwindows server, or by the procedure KillClient, which closes a connection to an Xwindows client.

**System action**  
Xwindows halts.

**Operator response**  
Notify the system programmer of the error.

**System programmer response**  
Check the Xwindows server to determine why it shut down or closed the client connection.

**Module**  
XLIBINT

**Procedure name**  
_XDefaultIOError

**EZB1211I**  
Request failed due to IUCV error.
Explanation
This message follows messages EZB1208E and EZB1209E, and indicates that the error displayed in those messages was caused by a failure in the inter-user communication vehicle (IUCV). Xwindows uses IUCV to transmit requests and receive replies.

System action
Xwindows halts.

Operator response
Notify the system programmer of the error.

System programmer response
See additional messages to determine the cause of the IUCV error and respond as indicated.

Module
XLIBINT

Procedure name
_XDefaultIOError

EZB1212E  Xlib: extension “extension” reason on display “display”.

Explanation
Xwindows encountered an incorrect event handler when the function XSetExtensionErrorHandler or XMissingExtension is called.

System action
Xwindows halts.

Operator response
Notify the system programmer of the error.

System programmer response
Make sure that the event handler is set correctly before calling the associated extension. For more information see the Xwindows documentation.

Module
EXTUTIL

Procedure name
XExtDisplayInfo

EZB1220I  name: argument, value: 0xhex_value

Explanation
This message displays the name and hexadecimal value of arguments for Xwindows processing.
**System action**
Xwindows continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
ARGLIST

**Procedure name**
PrintArgList

**EZB1221E**  
**error_prefix** Error: reason

**Explanation**
Xwindows has encountered an error. The reason for the error is described in the reason portion of this message.

**System action**
Xwindows halts.

**Operator response**
Notify the system programmer of the error.

**System programmer response**
Respond as indicated by the reason portion of this message.

**Module**
ERROR

**Procedure name**
_XtDefaultError

**EZB1222E**  
**warning_prefix** Warning: reason

**Explanation**
Xwindows has encountered an error. The reason for the error is described in the reason portion of this message.

**System action**
Xwindows halts.

**Operator response**
Notify the system programmer of the error.
System programmer response
Respond as indicated by the reason portion of this message.

Module
ERROR

Procedure name
_XtDefaultWarning

EZB1230I header w: children, m: request_mode, x: x_coordinate, y: y_coordinate, w: width, h: height, b: border_width

Explanation
This message displays information about the geometry of the current Xwindows.

System action
Xwindows continues.

Operator response
None.

System programmer response
None.

Module
GEOUTILS

Procedure name
PrintBox
Chapter 4. EZB2xxxx messages

EZB2000I  string string hexdata

Explanation
The contents of a packet or control block are displayed.

System action
The X25 server continues processing.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
SNAPIA5, SNAPAREA

EZB2010I  program {MVS} update level level

Explanation
The version of X25IPI currently running has the indicated program name (usually X25IPI) and the indicated update level.

System action
TCPIP continues.

Operator response
None.

System programmer response
Use the update level when reporting errors.

Module
XNX25IPI

Procedure name
None.

EZB2011I  Command: command
Explanation
This message is displayed when tracing is on. The X25 server is processing the command passed to it from the console input program.

System action
The command is processed and execution continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
COMMAND

EZB2012E    Unrecognized command: command

Explanation
The X25 server tried to process an unrecognized command that was passed to it from the console input program.

System action
The X25 server continues processing.

Operator response
Correct the command and resubmit. See z/OS Communications Server: IP Configuration Reference for more information about TCPIP X25 commands.

System programmer response
None.

Module
XNX25IPI

Procedure name
COMMAND

EZB2013T    Unrecognized CIB verb= cibverb

Explanation
The X25IPI server received an unexpected command verb in a command input buffer from the command queue (QEDIT). The valid commands are STOP, START, and MODIFY.
System action
X25IPI treats the command as STOP and exits.

Operator response
Restart TCPIPX25.

System programmer response
Report the problem to IBM software support services.

Module
XNX25IPI

Procedure name
COMMAND

**EZB2020I** MCH lu state state session_state

Explanation
X25IPI status for the indicated MCH is displayed. The following are the Link_state fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'00'</td>
<td>Restart needed</td>
</tr>
<tr>
<td>X'10'</td>
<td>Ready</td>
</tr>
<tr>
<td>X'20'</td>
<td>Restart request sent</td>
</tr>
<tr>
<td>X'30'</td>
<td>Restart indication received</td>
</tr>
</tbody>
</table>

The following are the VTAM session_state fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'20'</td>
<td>Enabled for LOGAPPL logon</td>
</tr>
<tr>
<td>X'30'</td>
<td>logon pending</td>
</tr>
<tr>
<td>X'40'</td>
<td>Opening</td>
</tr>
<tr>
<td>X'50'</td>
<td>Open (ready)</td>
</tr>
<tr>
<td>X'60'</td>
<td>Closing</td>
</tr>
<tr>
<td>X'70'</td>
<td>Unsuccessful</td>
</tr>
</tbody>
</table>

System action
TCPIP continues.
Operator response
None.

System programmer response
Use the status in problem determination.

Module
XNX25IPI

Procedure name
None.

**EZB2021I** VC vc LU lu DTE address state call_state session_state

Explanation
X25IPI status for the indicated virtual circuit is displayed. The following are the call_state fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'00'</td>
<td>Waiting for restart on MCH</td>
</tr>
<tr>
<td>X'10'</td>
<td>Available for connections</td>
</tr>
<tr>
<td>X'20'</td>
<td>Call request sent</td>
</tr>
<tr>
<td>X'30'</td>
<td>Call request received</td>
</tr>
<tr>
<td>X'40'</td>
<td>Ready for data transfer</td>
</tr>
<tr>
<td>X'41'</td>
<td>Reset request sent</td>
</tr>
<tr>
<td>X'42'</td>
<td>Reset request received</td>
</tr>
<tr>
<td>X'43'</td>
<td>Interrupt request sent</td>
</tr>
<tr>
<td>X'44'</td>
<td>Interrupt request received</td>
</tr>
<tr>
<td>X'50'</td>
<td>Call collision</td>
</tr>
<tr>
<td>X'60'</td>
<td>Clear request sent</td>
</tr>
<tr>
<td>X'70'</td>
<td>Clear request received</td>
</tr>
</tbody>
</table>

The following are the VTAM session_state fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'00'</td>
<td>Disabled until restart on MCH</td>
</tr>
</tbody>
</table>
Available for connections

Enabled for NPSI LU logon

Logon pending

Opening

Open (ready)

Closing

Failed

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

Use the status in problem determination.

**Module**

XNX25IPI

**Procedure name**

None.

**EZB2022R**  
**IP AS_asname state connection status**

**Explanation**

X25IPI status for the DLC path to the TCPIP address space is displayed. Possible values are:

X'80'

   DLC connection complete

X'40'

   DLC connection pending

X'00'

   no path

**System action**

The X25 server continues processing.

**Operator response**

Use the status in problem determination.
System programmer response
Use the status in problem determination.

Module
XNX25IPI

Procedure name
CLIST

EZB2030I  MCH lu RC: name IP: name SN: name TX: name

Explanation
In response to an EVENTS command, the internal subroutine names handling events associated with this MCH are displayed.

Subroutine
Description
IP
VTAM input pending
RC
VTAM request complete
SN
VTAM session notification
TX
Timer expired

System action
Processing continues.

Operator response
None.

System programmer response
Use the data for problem determination.

Module
XNX25IPI

Procedure name
None.

EZB2031I  VC vc RC: name IP: name SN: name TX: name

Explanation
In response to an EVENTS command, the internal subroutine names handling events associated with this virtual circuit is displayed.

Subroutine
Description
RC
VTAM request complete
**IP**
VTAM input pending

**SN**
VTAM session notification

**TX**
Timer expired

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Use the data for problem determination.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2032I**  MCH lu

**Explanation**
The following X25IPI014R VC messages apply to virtual circuits on this MCH.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2033I**  VC vc DTE_address S send_count R receive_count D drop_count Q queue_size

**Explanation**
The traffic counts for this virtual circuit are displayed.
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

---

**EZB2034I**  
IP AS_userid S send_count R receive_count D drop_count Q queue_size

**Explanation**
The traffic counts on the IUCV connection to the TCPIP address space are displayed.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

---

**EZB2040E**  
MCH lu unknown

**Explanation**
An attempt was made to send a call request for an unknown MCH. The request is ignored. The CERTCALL command is refused.

**System action**
TCPIP continues.

**Operator response**
Determine the correct MCH name, and reissue the CERTCALL command.
**System programmer response**
Assist the user as necessary.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2041I  MCH lu restarting**

**Explanation**
X25IPI is restarting this MCH in response to a RESTART command. A VTAM session is initiated for the NPSI MCH LU.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2042E  MCH lu unavailable**

**Explanation**
X25IPI could not restart this MCH because it is unavailable for logon. The MCH session is left available for initiation by LOGAPPL.

**System action**
TCPIP continues.

**Operator response**
Activate the MCH LU through VTAM.

**System programmer response**
Determine the reason why VTAM refused session initiation.

**Module**
XNX25IPI
Procedure name
None.

EJB2043E  MCH lu already started

Explanation
X25IPI could not restart this MCH because it is already active. The request is ignored.

System action
TCPIP continues.

Operator response
Notify the system programmer of the error.

System programmer response
Determine the correct MCH name and reissue the CERTCALL command.

Module
XNX25IPI

Procedure name
None.

EJB2050I  object at address additional

Explanation
This message reports the locations of internal storage as requested by debug flag 4. The object can be "Global storage", "MCH lu SDA", or "VC vc SDA". The additional variable, if present, reports the "stack at address limit address".

System action
The X25 server continues processing.

Operator response
None.

System programmer response
Use the storage addresses for problem determination.

Module
XNX25IPI

Procedure name
STRTLINK

EJB2051I  MCH lu SDA at address

Explanation
The contents of the session data area (SDA) for this connection are dumped.
System action
TCPIP continues.

Operator response
None.

System programmer response
Use the data for problem determination.

Module
XNX25IPI

Procedure name
None.

EZB2080I Dispatch handler name (event handler address) SDA session data area address ECB event control block

Explanation
This message is displayed when tracing is on. The X25 server scans for posted event control blocks and calls event handlers.

System action
The X25 server continues processing.

Operator response
None.

System programmer response
Use the dispatch trace in problem determination.

Module
XNX25IPI

Procedure name
MAINLOOP

EZB2081I Dispatch handler name (event handler address) SDA session data area address

Explanation
This message is displayed when tracing is on. The X25 server scans for posted event control blocks and calls an MCH event handler.

System action
The event handler is called.

Operator response
None.
System programmer response
Use the dispatch trace in problem determination.

Module
XNX25IPI

Procedure name
MCHEVENT

EZB2082I Main wait: number of ECBs ECBs

Explanation
This debug message displays the number of event control blocks (ECB) in the main wait list.

System action
The X25 server continues processing.

Operator response
None.

System programmer response
Use the event count in problem determination.

Module
XNX25IPI

Procedure name
MAINLOOP

EZB2083I Call caller's name (caller's address) from previous caller in previous caller's name (previous caller's address)

Explanation
This debug message displays the name of the current subroutine and the calling subroutine.

System action
The X25 server continues processing.

Operator response
None.

System programmer response
Use the trace in problem determination.

Module
XNX25IPI
**Procedure name**
FOLLOW

**EZB2084E**  Posted ECB at *address* had no handler

**Explanation**
The X25IPI event control block at this address did not specify an event handler routine.

**System action**
TCPIP continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
If the error recurs, obtain a dump of the X25IPI address space to make sure that the correct event handler is loaded.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2085E**  ECB address being added twice to wait list

**Explanation**
The X25IPI routine that adds elements to the ECB wait list detected an attempt to add an ECB twice.

**System action**
TCPIP continues.

**Operator response**
Notify the system programmer about the error.

**System programmer response**
Determine the external event that caused this condition. Re-create it with an X25 internal trace running. Obtain a dump of the X25 region and submit the dump to IBM software support services.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2090I**  Terminating

**Explanation**
X25IPI is terminating its execution and shutting down.
System action
The VTAM ACB is closed.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

EZB2091I  HALT notice accepted type type

Explanation
X25IPI received a VTAM HALT notification or a HALT console command and is shutting down execution.

System action
VTAM LU sessions are closed, and the IUCV connection to the TCPIP address space is severed.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

EZB2092T  Stack overflow at address

Explanation
X25IPI encountered a stack overflow at the indicated address. ABEND message X'091' is displayed.

System action
X25IPI abends.

Operator response
Tell the system programmer about the error.
System programmer response
Determine the cause of overflow, and submit the ABEND dump to IBM software support services.

Module
XNX25IPI

Procedure name
None.

EZB2093T  Buffer released twice at address in routine (address)

Explanation
X25IPI encountered a consistency error in its buffer allocation pool. A buffer was released twice. This indicates a programming error. Return code X'92'. is displayed.

System action
X25IPI abends.

Operator response
Tell the system programmer about the error.

System programmer response
Determine the cause of the consistency error, and submit the ABEND dump to IBM software support services.

Module
XNX25IPI

Procedure name
None.

EZB2094S  NPSI SEND completion, pending packet = packet number

Explanation
A program flag indicating a deferred control packet to be sent has an unacceptable value. The pending condition is reset.

System action
TCPIP continues.

Operator response
Tell the system programmer about the error.

System programmer response
Contact IBM software support services.

Module
XNX25IPI
Procedure name
None.

**EZB2095T**  WTO text overflow

**Explanation**
XNX25IPI encountered an overflow in the Write To Operator (WTO) area, which is used to display informational and error messages. ABEND message X'099' is displayed.

**System action**
The program abends.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Determine the cause of the overflow, and submit the ABEND dump to IBM software support services.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2099I**  Ended

**Explanation**
X25IPI has finished processing.

**System action**
The X25IPI program exits.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2100I**  configuration dataset record

**Explanation**
This debug message displays a record that was read from the configuration data set.
System action
The X25 server continues processing.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
X25IPI1

EZB2101E Unable to open configuration file, DDNAME=X25IPI

Explanation
X25IPI encountered an error opening its X25IPI configuration data set.

System action
X25IPI does not start.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the DD for the X25IPI configuration data set.

Module
XNX25IPI

Procedure name
None.

EZB2102E Unrecognized configuration entry: keyword

Explanation
X25IPI encountered an unrecognized entry in its X25IPI configuration data set. This configuration record is skipped; further configuration errors can result.

System action
Processing continues.

Operator response
Tell the system programmer about the error.
System programmer response
Correct the entry in the X25IPI configuration data set.

Module
XNX25IPI

Procedure name
None.

EZB2103E  Missing configuration entry: keyword

Explanation
X25IPI could not find the indicated keyword in its configuration data set. You should have at least one entry defining a link in the file.

System action
X25IPI does not start.

Operator response
Tell the system programmer about the error.

System programmer response
Add a Link definition to the X25IPI configuration data set for each indicated keyword to be used for TCPIP traffic.

Module
XNX25IPI

Procedure name
None.

EZB2104E  Buffer storage not available (bytes bytes required)

Explanation
X25IPI could not allocate sufficient storage during initialization.

System action
X25IPI does not start.

Operator response
Tell the system programmer about the error.

System programmer response
Increase the X25IPI region or address space storage size to the indicated number of bytes.

Module
XNX25IPI
**EZB2106E**  DLC Init function X25IPI failed R15=value

**Explanation**
X25IPI encountered an error issuing a DLC Init call.

**System action**
X25IPI does not start.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Verify that the X25IPI procedure is in the PROCLIB.

**Module**
XNX25IPI

**EZB2107T**  Programming error: not enough buffers allocated

**Explanation**
The calculation of the number of buffers to allocate was incorrect.

**System action**
X25IPI does not start.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Contact IBM software support services.

**Module**
XNX25IPI

**EZB2108I**  Unable to obtain TCPIPJobname from TCPIP.DATA, default TCPIP job name used

**Explanation**
X25IPI encountered an error obtaining the TCP/IP stack name from the TCPIP.DATA file. The default job name of TCPIP is being used.
System action
TCPIP continues.

Operator response
If TCP connections cannot be made, notify the system programmer.

System programmer response
If the job name of the TCPIP stack to be used by X25 is not TCPIP use the TCPIP.DATA TCPIPJOBNAME statement to specify the correct name. See z/OS Communications Server: IP Configuration Guide for how TCPIP.DATA statements are searched for.

Module
XNX25IPI

Procedure name
None.

EZB2110E   Unrecognized trace level: level

Explanation
The trace level, specified in either the X25IPI configuration data set or the TRACE command, is not correct. The trace level is set to “off”.

System action
TCPIP continues.

Operator response
Reissue the TRACE command with valid trace level.

System programmer response
Correct the Trace entry in the X25IPI configuration data set.

Module
XNX25IPI

Procedure name
None.

EZB2111I   VTAM ACB IPI_APPN opened successfully

Explanation
The virtual transmission access method (VTAM) address space successfully opened an activity control block (ACB) for the indicated application.

System action
The application begins processing. TCPIP continues.
Operator response
None.

System programmer response
None.

Module
xnx25ipi

Procedure name
main

**EZB2112E**  Repeated VTAM record ignored: record

Explanation
X25IPI encountered more than one VTAM entry in the X25IPI configuration data set. The repeated VTAM record is skipped.

System action
Initialization continues.

Operator response
Tell the system programmer about the error.

System programmer response
Remove the repeated VTAM record.

Module
XNX25IPI

Procedure name
None.

**EZB2113E**  VTAM application name missing

Explanation
The VTAM application ID is missing on the VTAM entry in the X25IPI configuration data set.

System action
X25IPI does not start.

Operator response
Tell the system programmer about the error.

System programmer response
Specify the VTAM application ID and password on the VTAM entry.
**Module**
XNX25IPI

**Procedure name**
None.

**EZB2114E**  VTAM application password missing

**Explanation**
The VTAM application password is missing on the VTAM entry in the X25IPI configuration data set.

**System action**
X25IPI does not start.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Specify the VTAM application password on the VTAM entry.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2115E**  VTAM ACB application open failed

**Explanation**
X25IPI encountered an error opening the VTAM ACB. This message is preceded by a EZB2401E or EZB2402E message reporting the VTAM error code.

**System action**
X25IPI does not start.

**Operator response**
Activate the X25IPI application in VTAM.

**System programmer response**
Use the VTAM error code from the message EZB2401E or EZB2402E to determine why the VTAM ACB OPEN macro was unsuccessful.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2121E**  DDN and non-DDN links cannot be mixed
**Explanation**

X25IPI encountered Link statements in its X25IPI configuration data set specifying both DDN and non-DDN links. The Link entry is discarded; further errors can result.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Separate DDN and non-DDN links into 2 X25IPI virtual machines.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2122E**  Link LU name missing

**Explanation**
The NPSI MCH LU name is missing on the Link entry in the X25IPI configuration data set. The Link entry is discarded; further errors can result.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Complete the Link entry. See z/OS Communications Server: IP Configuration Guide for the format of the Link definition.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2123E**  Link DTE address missing

**Explanation**
X25IPI encountered a Link statement without a DTE address specified. This field is required for all non-DDN links. The Link entry is discarded; further errors can result.
**System action**  
Processing continues.

**Operator response**  
Tell the system programmer about the error.

**System programmer response**  
Complete the Link entry. See z/OS Communications Server: IP Configuration Guide for the format of the Link definition.

**Module**  
XNX25IPI

**Procedure name**  
None.

---

**EZB2124E**  
**Link window size missing or not numeric**

**Explanation**  
X25IPI encountered a Link statement in its X25IPI configuration data set that had a missing or nonnumeric default window size specification. The Link entry is discarded; further errors can result.

**System action**  
Processing continues.

**Operator response**  
Tell the system programmer about the error.

**System programmer response**  
Correct the window size. See z/OS Communications Server: IP Configuration Reference for the format of the Link definition.

**Module**  
XNX25IPI

**Procedure name**  
None.

---

**EZB2125E**  
**Link window size not in range 1..7 or 1..127**

**Explanation**  
X25IPI encountered a Link statement in its X25IPI configuration data set with an incorrect default window size specification. The window size should be in the range 1–7 or 1–127. The Link entry is discarded; further errors can result.

**System action**  
Processing continues.
Operator response
Tell the system programmer about the error.

System programmer response
Correct the window size.

Module
XNX25IPI

Procedure name
None.

EZB2126E  Link packet size missing or not numeric

Explanation
X25IPI encountered a Link statement in its X25IPI configuration data set that had a missing or nonnumeric default packet size. The Link entry is discarded; further errors can result.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the packet size. See z/OS Communications Server: IP Configuration Guide for the format of the Link definition.

Module
XNX25IPI

Procedure name
None.

EZB2127E  Link packet size unacceptable

Explanation
X25IPI encountered a Link statement in its X25IPI configuration data set with an incorrect default packet size specification. The size should be 32, 64, 128, 256, 512, 1024, 2048, or 4096. The Link entry is discarded; further errors can result.

System action
The link entry is discarded. Processing continues.

Operator response
Tell the system programmer about the error.
**System programmer response**
Correct the packet size.

**Module**
XNX25IPI

**Procedure name**
None.

**EJB2128E**  Link logical channel count missing or not numeric

**Explanation**
X25IPI encountered a Link statement in its X25IPI configuration data set that had a missing or nonnumeric logical channel count field. The Link entry is discarded; further errors can result.

**System action**
The Link entry is discarded. Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Correct the channel count. See z/OS Communications Server: IP Configuration Guide for the format of the Link definition.

**Module**
XNX25IPI

**Procedure name**
None.

**EJB2129E**  Link logical channel count not in range 1..1023

**Explanation**
X25IPI encountered a Link statement in its X25IPI configuration data set with an incorrect logical channel count field. The count should be in the range 1–1023. The Link entry is discarded; further errors can result.

**System action**
The Link entry is discarded. Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Correct the channel count.

**Module**
XNX25IPI
**EZB2130E**  Link reserved channel count missing or not numeric

**Explanation**
X25IPI encountered a Link statement in its X25IPI configuration data set that had a missing or nonnumeric reserved channel count. The Link entry is discarded; further errors can result.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Correct the reserved channel count. See z/OS Communications Server: IP Configuration Guide for the format of the Link definition.

**Module**
XNX25IPI

---

**EZB2131E**  Link reserved channel count not in range 1..LCC

**Explanation**
X25IPI encountered a Link statement in its X25IPI configuration data set with an incorrect reserved channel count. The count should be between 1 and the number of logical channels that have been defined. The Link entry is discarded; further errors can result.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Correct the reserved channel count. See z/OS Communications Server: IP Configuration Guide for the format of the Link definition.

**Module**
XNX25IPI

---

**EZB2132E**  Link data network code missing
Explanation
X25IPI encountered a Link statement in its X25IPI configuration data set that omitted a data network identifier code (DNIC). DNICs should be specified for all networks (use DDN for DDN nets). The Link entry is discarded; further errors can result.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Complete the Link entry. See z/OS Communications Server: IP Configuration Guide for the format of the Link definition.

Module
XNX25IPI

Procedure name
None.

EZB2133E   Data network identifier code not decimal

Explanation
X25IPI encountered a Link statement in its X25IPI configuration data set that specified a nondecimal data network identifier code (DNIC). The DNIC should be decimal. The Link entry is discarded; further errors can result.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the DNIC number. See z/OS Communications Server: IP Configuration Guide for the format of the Link definition.

Module
XNX25IPI

Procedure name
None.

EZB2134E   Link DTE address not decimal

Explanation
X25IPI encountered a Link statement in its X25IPI configuration data set that contained a nondecimal DTE address. The Link entry is discarded; further errors can result.
System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the DTE address.

Module
XNX25IPI

Procedure name
None.

EZB2135E  Altlink record not preceded by a Link record

Explanation
The X25 server is processing an ALTLINK record that was not preceded by a LINK record in the configuration data set.

System action
X25 ignores the ALTLINK configuration record.

Operator response
Correct the configuration data set by either removing the ALTLINK statement or adding the correct LINK statement preceding the ALTLINK statement. For more information about the ALTLINK and LINK statements, see z/OS Communications Server: IP Configuration Guide.

System programmer response
None.

Module
XNX25IPI

Procedure name
STRTALTL

EZB2140E  Dest record not preceded by Link record

Explanation
X25IPI encountered a Dest record in its X25IPI configuration data set before any Link record. The Dest record should follow the corresponding Link record in the X25IPI configuration data set. The Dest entry is skipped.

System action
Processing continues.
Operator response
Tell the system programmer about the error.

System programmer response
Move the Dest record after the corresponding Link record.

Module
XNX25IPI

Procedure name
None.

EZB2141E  Dest IP address missing

Explanation
X25IPI encountered a Dest record in its X25IPI configuration data set that did not specify an IP address. The Dest entry is discarded.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Complete the Dest entry. See z/OS Communications Server: IP Configuration Guide for the format of the Dest record.

Module
XNX25IPI

Procedure name
None.

EZB2142E  Dest IP address must be decimal

Explanation
X25IPI encountered a Dest record in its X25IPI configuration data set specifying a nondecimal destination address. The destination address should be in dotted-decimal form. The Dest entry is discarded.

System action
Processing continues.

Operator response
Tell the system programmer about the error.
**System programmer response**
Correct the IP address. See z/OS Communications Server: IP Configuration Guide for the format of the Dest record.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2143E  Dest DTE address missing**

**Explanation**
X25IPI encountered a Dest record in its X25IPI configuration data set, which omitted the destination X.25 DTE address specification. The destination X.25 DTE address specification is required on non-DDN networks. The Dest entry is discarded.

**System action**

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Complete the Dest record with the X.25 DTE address. See z/OS Communications Server: IP Configuration Guide for the format of the Dest record.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2144E  Dest DTE address must be decimal**

**Explanation**
X25IPI encountered a Dest record in its X25IPI configuration data set that had a nondecimal destination X.25 DTE address specified. The destination X.25 DTE address should be decimal. The Dest entry is discarded.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Correct the DTE address.
Module
XNX25IPI

Procedure name
None.

EZB2145E Dest call user data must be hexadecimal

Explanation
X25IPI encountered a Dest record in its X25IPI configuration data set specifying nonhexadecimal call user data (CUD) protocol ID. The CUD should be hexadecimal. The Dest entry is discarded.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the call user data.

Module
XNX25IPI

Procedure name
None.

EZB2146E Dest facilities data must be hexadecimal

Explanation
The X25 server is processing the facilities field on a DEST statement. The field contained a non-hexadecimal character.

System action
X25 discards this record and continues.

Operator response
Correct the configuration data set by correcting the DEST facilities field. For more information about the configuration data set statements, see z/OS Communications Server: IP Configuration Reference.

System programmer response
None.

Module
XNX25IPI

Procedure name
STRTDEST

208 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**EZB2150E  Datagram size limit missing or not numeric**

**Explanation**

X25IPI encountered a Buffers record in its X25IPI configuration data set that had a missing or nonnumeric buffer size specification. The Buffers record is ignored. The default values are applied.

**System action**

Processing continues.

**Operator response**

Tell the system programmer about the error.

**System programmer response**

Complete the Buffers record with the datagram size limit. See z/OS Communications Server: IP Configuration Guide for the format of the Dest record.

**Module**

XNX25IPI

**Procedure name**

None.

---

**EZB2151E  Datagram size limit not in range 576...2048**

**Explanation**

X25IPI encountered a Buffers record in its X25IPI configuration data set with an incorrect buffer size specification. The buffer size should be in the range 576–2048. The Buffers record is ignored. The default values are applied.

**System action**

Processing continues.

**Operator response**

Tell the system programmer about the error.

**System programmer response**

Correct the datagram size limit.

**Module**

XNX25IPI

**Procedure name**

None.

---

**EZB2152E  Extra buffer count missing or not numeric**

**Explanation**

X25IPI encountered a Buffers record in its X25IPI configuration data set that had a missing or nonnumeric buffer count field. The extra buffer count is ignored.
System action
The extra buffer count is ignored. Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the buffer count.

Module
XNX25IPI

Procedure name
None.

EZB2153E  VC send queue limit missing or not numeric

Explanation
X25IPI encountered a Buffers record in its X25IPI configuration data set that had a missing or nonnumeric send queue limit. The send queue limit defaults to 8.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the queue limit.

Module
XNX25IPI

Procedure name
None.

EZB2155E  Inactivity timeout missing or not numeric

Explanation
X25IPI encountered a Timers record in its X25IPI configuration data set that had a missing or nonnumeric inactivity time-out value. The Timers record is ignored. The default values are applied.

System action
Processing continues.

Operator response
Tell the system programmer about the error.
System programmer response
Complete the Timers record with the inactivity time-out. See z/OS Communications Server: IP Configuration Reference for the format of the Dest record.

Module
XNX25IPI

Procedure name
None.

EZB2156E Minimum call timer missing or not numeric

Explanation
X25IPI encountered a Timer record in its X25IPI configuration data set that had a missing or nonnumeric minimum time field. This Timer record is ignored. The minimum call time defaults to 60 seconds.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the minimum call time.

Module
XNX25IPI

Procedure name
None.

EZB2160E Options record not preceded by Link record

Explanation
X25IPI encountered an Options record in its X25IPI configuration data set before any Link record. The Options record should follow the associated Link record. The Options record is ignored.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Move the Options record after the corresponding Link record.

Module
XNX25IPI
EZB2161E  Option name not recognized name

Explanation
X25IPI encountered an Options record in its X25IPI configuration data set with an unrecognized option name. The remainder of the Options record is skipped.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the option name. See z/OS Communications Server: IP Configuration Guide for the format of the Options record.

Module
XNX25IPI

Procedure name
None.

EZB2162E  Option packet size missing or not numeric

Explanation
X25IPI has encountered an Options statement in its X25IPI configuration data set (DD statement X25IPI) that has a missing or nonnumeric packet size specification. The remainder of the Options record is skipped.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the packet size.

Module
XNX25IPI

Procedure name
None.

EZB2163E  Option packet size unacceptable
Explanation
X25IPI encountered an Options record in its X25IPI configuration data set that specified an unacceptable packet size. The size should be 32, 64, 128, 256, 512, 1024, 2048, or 4096. The remainder of the Options record is skipped.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the packet size.

Module
XNX25IPI

Procedure name
None.

EZB2164E Option window size missing or not numeric

Explanation
XNX25IPI encountered an Options statement in its X25IPI configuration data set that had a missing or nonnumeric window size. The remainder of the Options record is skipped.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the window size.

Module
XNX25IPI

Procedure name
None.

EZB2165E Option window size not in range 1..7 or 1..127

Explanation
XNX25IPI encountered an Options statement in its X25IPI configuration data set that specified an incorrect window size. The window size should be in the range 1–7 or 1–127. The remainder of the Options record is skipped.
System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the window size.

Module
XNX25IPI

Procedure name
None.

EZB2166E   Option call user data missing or not hexadecimal

Explanation
The X25 server is processing an OPTION CALLDATA statement. The statement is missing the call user data, or the call user data specified was not hexadecimal.

System action
X25 discards this record and continues.

Operator response
Correct the configuration data set by correcting the CALLDATA option. For more information about the configuration data set statements, see z/OS Communications Server: IP Configuration Guide.

System programmer response
None.

Module
XNX25IPI

Procedure name
STRTOPTN

EZB2167E   Option facilities data must be hexadecimal

Explanation
The X25 server is processing an OPTION FACILITIES statement. The statement is missing the facilities data, or the facilities data specified was not hexadecimal.

System action
X25 discards this record and continues.
Operator response
Correct the configuration data set by correcting the FACILITIES option. For more information about the configuration data set statements, see z/OS Communications Server: IP Configuration Reference.

System programmer response
None.

Module
XNX25IPI

Procedure name
STRTOPTN

EZB2180E  FAST record not preceded by a Link record

Explanation
The X25 server is processing the FAST statement in the configuration data set. The FAST statement was not preceded by a LINK statement.

System action
XNX25IPI ignores the FAST statement.

Operator response
Correct the configuration data set by either removing the FAST statement or preceding the FAST statement with a LINK statement. Rerun the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see z/OS Communications Server: IP Configuration Guide.

System programmer response
None.

Module
XNX25IPI

Procedure name
STRTFAST

EZB2181E  FAST connect LU name prefix missing

Explanation
The X25 server is processing the FAST statement in the configuration data set. The FAST statement did not contain a VC LU prefix.

System action
XNX25IPI ignores the FAST statement.

Operator response
Correct the configuration data set by adding the prefix to the FAST statement. Rerun the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see z/OS Communications Server: IP Configuration Guide.
**EZB2182E**  FAST connect LU name too long

**Explanation**
The X25 server is processing the FAST statement in the configuration data set. The FAST statement contained a VC LU prefix that exceeded eight characters.

**System action**
XNX25IPI ignores the FAST statement.

**Operator response**
Correct the configuration data set by correcting the prefix in the FAST statement. Rerun the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see z/OS Communications Server: IP Configuration Guide.

**EZB2183E**  FAST connect LU name suffix unacceptable

**Explanation**
The X25 server is processing the FAST statement in the configuration data set. The FAST statement contained a suffix that had an incorrect character. The suffix should be decimal or hexadecimal, as appropriate to the specified FAST connect numbering scheme.

**System action**
Remaining fast-connect LU names will not be generated.

**Operator response**
Correct the configuration data set by correcting the suffix in the FAST statement. Rerun the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see z/OS Communications Server: IP Configuration Guide.

**System programmer response**
None.
**EJB2184E  FAST connect LU name suffix overflow**

**Explanation**
The X25 server is processing the FAST statement in the configuration data set. The FAST statement contains a suffix that has too many characters. The LU name suffix generated for a fast-connect virtual circuit exceeds the available field width.

**System action**
Remaining fast-connect LU names will not be generated.

**Operator response**
Correct the configuration data set by correcting the suffix in the FAST statement. Rerun the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see z/OS Communications Server: IP Configuration Guide.

**System programmer response**
None.

---

**EJB2201I  MCH luname OPNDST complete**

**Explanation**
X25IPI has successfully established a VTAM session with the NPSI MCH LU.

**System action**
The X25 server continues processing.

**Operator response**
None.

**System programmer response**
None.
**EZB2202I**  MCH **luname** restart packet sent

**Explanation**
The X25 server has sent a restart packet to a multichannel link (MCH).

**System action**
The contents of the restart packet are displayed in subsequent messages.

**Operator response**
None.

**System programmer response**
Use the data in problem determination.

**Module**
XNX25IPI

---

**EZB2203I**  MCH **lu** restarting

**Explanation**
An X.25 restart exchange was started on the NPSI MCH.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

---

**EZB2205E**  MCH **lu** open failed

**Explanation**
X25IPI encountered an error opening the indicated NPSI MCH. This message is preceded by the message EZB2407E or EZB2411E reporting a VTAM error code. The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.
System action
Processing continues.

Operator response
Activate the NPSI MCH in VTAM, and use the X25IPI RESTART command to reacquire the MCH.

System programmer response
Use the VTAM error code from the EZB2407E or EZB2411E message to determine why the MCH OPNDST was unsuccessful.

Module
XNX25IPI

Procedure name
None.

EZB2206E  MCH lu OPNDST did not complete

Explanation
A VTAM OPNDST request on a NPSI MCH LU was posted complete, but the NIB is not marked open. The contents of the session data area (SDA) for this MCH are dumped. The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.

System action
Processing continues.

Operator response
Activate the NPSI MCH in VTAM, and use the X25IPI RESTART command to reacquire the MCH.

System programmer response
Determine the state of the NPSI MCH LU using the VTAM DISPLAY command.

Module
XNX25IPI

Procedure name
None.

EZB2210I  MCH luname packet level ready

Explanation
The X25 server has made a multichannel link (MCH) available for virtual circuit connections.

System action
XNX25IPI continues processing.

Operator response
None.
System programmer response
None.

Module
XNX25IPI

Procedure name
MCHOPNDC

EZB2211I  MCH lu name packet received

Explanation
The X25 server has received a packet from a multichannel link (MCH).

System action
The contents of the packet are displayed in subsequent messages.

Operator response
None.

System programmer response
Use the data in problem determination.

Module
XNX25IPI

Procedure name
MCHRSTIP

EZB2212I  MCH lu discarded packet during restart

Explanation
X25IPI received an incorrect packet while restarting this MCH. The incorrect packet is dumped and discarded.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI
**EZB2213I**  
**MCH lu restart indication, cause=value diagnostic=value**

**Explanation**
X.25IPI received an X.25 restart indication for this MCH during the restart procedure with the indicated cause and diagnostic bytes. The X.25 network interface has been reinitialized. Virtual circuit connections on the MCH are closed, and the X.25 restart procedure is used to place the MCH back in operation. See *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning* for a list of X.25 cause and diagnostic codes. See the provider of the X.25 network service for documentation that lists additional diagnostic codes.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2214I**  
**MCH lu restart confirmation**

**Explanation**
X.25IPI received a restart confirmation for this MCH. The MCH is marked ready for new connections.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2215I**  
**MCH lu restart complete**
**Explanation**
X25IPI has completed terminating virtual circuits on an MCH undergoing restart. An X.25 restart confirmation is sent to complete the restart procedure.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

---

**EZB2221I**  MCH lu restart indication, cause=cause, diagnostic=diagnostic

**Explanation**
X25IPI received a restart indication on this MCH with the indicated cause and diagnostic bytes. The X.25 network interface has been reinitialized. Virtual circuit connections on the MCH are closed, and the X.25 restart procedure is used to place the MCH back in operation. See *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning* for a list of X.25 cause and diagnostic codes. See the provider of the X.25 network service for documentation that lists additional diagnostic codes.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

---

**EZB2222I**  MCH lu restart confirm packet sent

**Explanation**
X25IPI transmitted a restart confirmation packet for this MCH. The restart confirmation packet is dumped.
**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

---

**EZB2223I**  
**MCH luname request packet sent**

**Explanation**
The X25 server has sent a request packet from a multichannel link (MCH).

**System action**
The contents of the restart packet are displayed in subsequent messages.

**Operator response**
None.

**System programmer response**
Use the data in problem determination.

**Module**
XNX25IPI

**Procedure name**
MCSTSR SR

---

**EZB2224I**  
**MCH lu restarting**

**Explanation**
X25IPI is restarting the indicated MCH.

**System action**
X25IPI waits to receive a restart confirmation on the MCH.

**Operator response**
None.
**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2230I MCH lu name packet received**

**Explanation**
The X25 server has received a packet from a multichannel link (MCH).

**System action**
The contents of the restart packet are displayed in subsequent messages.

**Operator response**
None.

**System programmer response**
Use the data in problem determination.

**Module**
XNX25IPI

**Procedure name**
MCHRDYIP

**EZB2231I MCH lu orphan packet received**

**Explanation**
X25IPI received a packet for which it could find no associated connection. An X.25 clear request is sent on the virtual circuit.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI
**EZB2232I**  MCH lu diagnostic packet

**Explanation**
X25IPI received an X.25 diagnostic packet on this MCH. The diagnostic packet is dumped and discarded.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Use the diagnostic packet to obtain additional information about X.25 network errors.

**Module**
XNX25IPI

---

**EZB2233I**  MCH lu clear request sent

**Explanation**
X25IPI sent a clear request on the indicated MCH to recover from an error situation.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

---

**EZB2234E**  MCH lu no free path available for incoming call MCH lu check number of logical channels on Link record
**Explanation**

X25IPI received an incoming call, but has no session areas available to handle it. This indicates that there are more virtual circuits subscribed than were specified on the LINK record in the X25IPI configuration data set. The incoming call is cleared.

**System action**

Processing continues.

**Operator response**

Tell the system programmer about the error.

**System programmer response**

Increase the virtual circuit count on the Link record to match the number of virtual circuits defined in the NPSI configuration.

**Module**

XNX25IPI

**Procedure name**

None.

**EJB2235W**  

MCH lu orphan packet discarded

**Explanation**

X25IPI received a packet that it could not associate with any connection. The packet is dumped and discarded.

**System action**

Processing continues.

**Operator response**

Tell the system programmer about the error.

**System programmer response**

Determine the type of X.25 packet from the dump. A clear confirmation (X'17') can be discarded harmlessly.

**Module**

XNX25IPI

**Procedure name**

None.

**EJB2236E**  

MCH lu unrecognized packet received

**Explanation**

X25IPI did not recognize the type code in a packet received from NPSI. The unrecognized packet is dumped and discarded.
System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Determine the packet type from the dump; contact IBM software support services.

Module
XNX25IPI

Procedure name
None.

EZB2250I    MCH lu terminating

Explanation
X25IPI is terminating this MCH session in response to an error condition or a HALT command.

System action
Connections on virtual circuits associated with the MCH are terminated, and the MCH session is closed.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

EZB2251I    MCH lu closed

Explanation
X25IPI received a CLSDST completion indication for this MCH, indicating that this MCH has closed.

System action
The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.

Operator response
None.
**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2252I**  
MCH luname logon

**Explanation**
The multichannel link (MCH) was inactive and a logon notification was received. XNX25IPI opens a VTAM session.

**System action**
XNX25IPI continues processing.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
MCHRDYIP

**EZB2280E**  
MCH lu session loss code code

**Explanation**
X25IPI received the indicated MCH status change code while the MCH was not operational. See message EZB2285E for the loss codes.

**System action**
The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.

**Operator response**
Notify the system programmer of the error.

**System programmer response**
Manually restart X25IPI, if necessary.
**Procedure name**
None.

**EZB2281I**  VC vc packet discarded on failed session

**Explanation**
X25IPI discarded a packet received after a session was closed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2282E**  MCH lu unexpected request completion

**Explanation**
X25IPI received a VTAM request completion notification for this MCH. This indicates a program error, because MCH sends are done synchronously. The notification is ignored.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Contact IBM software support services.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2283I**  MCH lu DATE error report command=command error=error
**Explanation**

X25IPI received a NPSI Dedicated Access to X.25 Transport Extension (DATE) error report for the indicated command and error. See *X.25 Network Control Program Packet Switching Interface Host Programming* for the error codes.

**System action**

X25IPI attempts to recover from the error.

**Operator response**

None.

**System programmer response**

None.

**Module**

XNX25IPI

**Procedure name**

None.

**EZB2284E**

MCH lu DATE error report command=command error=error

**Explanation**

X25IPI encountered a DATE error because of an incorrect logical channel number. X25IPI discards the error indication. See *X.25 Network Control Program Packet Switching Interface Host Programming* for the error codes.

**System action**

Processing continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

XNX25IPI

**Procedure name**

None.

**EZB2285E**

MCH lu session loss code code

**Explanation**

X25IPI received a session status notification with the indicated session loss code for this MCH.

**Code**

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
</table>
Less than 50
   VTAM LOSTERM exit codes
50
   VTAM SCIP exit UNBIND
64000
   VTAM NS exit CLEANUP
64001
   VTAM NS exit session initiation failure
64002
   VTAM NS exit session initiation negative response

The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.

System action
Processing continues.

Operator response
Reactivate the NPSI MCH in VTAM. Use the X25IPI RESTART command to reacquire the MCH LU.

System programmer response
Use the VTAM error code to determine the reason why the session was unsuccessful.

Module
XNX25IPI

Procedure name
None.

EZB2301I    VC ID incoming call from address user data value

Explanation
X25IPI has received an incoming call on this virtual circuit from the indicated address with the indicated user data (protocol ID). The call is accepted or refused.

System action
Processing continues.

Operator response
None.

System programmer response
Use the address to determine the source of the connection.

Module
XNX25IPI

Procedure name
None.
EZB2302I  VC vc call accept packet sent

Explanation
X25IPI accepted an incoming call on this virtual circuit.

System action
The contents of the X.25 call accept packet are displayed in subsequent messages.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCINCOM

EZB2304W  VC vc incoming call cleared: caller not known

Explanation
X25IPI received a call on this virtual circuit from an address not present in the Dest entries for the associated MCH. The incoming call is cleared.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Determine the identity of the remote system initiating the call request, and add the address to the Dest list if the connection is authorized. The X.25 address of the calling system was noted in the preceding EZB2301I message for the VC ID.

Module
XNX25IPI

Procedure name
None.

EZB2305W  VC vc incoming call cleared: draining

Explanation
X25IPI cleared an incoming call because a VTAM HALT request has been issued to end communication. Incoming calls are refused once the VTAM HALT command is issued.
System action
Processing continues.

Operator response
Shutdown and restart X25IPI when VTAM is restarted.

System programmer response
Assist the user as necessary.

Module
XNX25IPI

Procedure name
None.

EZB2306E VC vc incoming call cleared: reason

Explanation
X25IPI cleared an incoming call on this virtual circuit because of an error in the format of the X.25 call request packet:
- Called address is not decimal.
- Calling address is not decimal.
- CUD field is too long.
- CUD field not acceptable.
- Duplicate address.
- Facilities not acceptable.
- Reverse charging has not been enabled on the associated link.
- Reverse charging refused.
- Reverse charging was specified in the call.
- Reverse charging has not been enabled on the associated link.
- The address duplicated that of another connection.

The incoming call is cleared.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the remote system that is generating the incorrect calls. The X.25 address of the calling system was noted in the preceding EZB2301I message for the VC ID.

Module
XNX25IPI
EJB2307I  VC vc clear request packet sent

Explanation
X25IPI refused an incoming call on this virtual circuit.

System action
The contents of the X.25 clear request packet are displayed in subsequent messages.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCINCOM

EJB2308I  VC vc finished sending call confirm

Explanation
A VTAM OPNDST call was accepted and the NPSI SEND completed.

System action
XNX25IPI continues processing.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCACPTSC

EJB2310I  VC vc outgoing call to address

Explanation
X25IPI is placing an outgoing call to this indicated address on this virtual circuit. Queued datagrams are sent on the connection after the call is accepted by the remote system.
System action
Processing continues.

Operator response
None.

System programmer response
Use the address to determine the destination of the connection.

Module
XNX25IPI

Procedure name
None.

EZB2311I VC vc call request packet sent

Explanation
X25IPI placed an outgoing call on this virtual circuit.

System action
The contents of the X.25 call request packet are displayed in subsequent messages.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCCLLFCL

EZB2312I VC vc call request sent

Explanation
X25IPI placed an outgoing call request on this virtual circuit. The X.25 NCP Packet Switching Interface (NPSI) SEND is complete.

System action
XNX25IPI continues processing.

Operator response
None.
System programmer response
None.

Module
XNX25IPI

Procedure name
VCCALLSC

EZB2313W  VC vc call timer expired

Explanation
The remote system has not responded to a call request in 200 seconds. The call is ended, and queued datagrams are discarded.

System action
Processing continues.

Operator response
Check the status of the remote system and the X.25 network. The X.25 address of the calling system was noted in the preceding EZB2310I message for the VC ID.

System programmer response
Assist the user as necessary.

Module
XNX25IPI

Procedure name
None.

EZB2314I  VC vc call accepted by address user data value

Explanation
An outgoing call on this virtual circuit was accepted by the remote system at the indicated address with the indicated user data (protocol specifier). Queued datagrams are sent.

System action
Processing continues.

Operator response
None.

System programmer response
Use the address to determine the destination of the connection.

Module
XNX25IPI
**EJB2315I  VC vc retrying call with packet size size**

**Explanation**
The packet size for this virtual circuit was reduced by the network or responder. NPSI DATE cannot handle this negotiation, thus the call is cleared and placed again with the smaller packet size.

**System action**
Processing continues.

**Operator response**
Tell the system programmer if this message recurs frequently.

**System programmer response**
Add or change an OPTIONS PACKETSIZE entry on the associated Link specifying the smallest packet size used by other systems on the X.25 network. Consider using the NPSI GATE facility rather than DATE. GATE can handle the reduced packet size without the call to be repeated.

**Module**
XNX25IPI

**EJB2316E  VC vc outgoing call cleared: reason**

**Explanation**
X25IPI cleared an outgoing call on this virtual circuit because of an error in one of the following formats:
- The X.25 call accept packet.
- Accepting user data is too long.
- Called address is not decimal.
- Calling address is not decimal.
- Facilities not acceptable.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Correct the remote system that is generating the incorrect calls. The X.25 address of the called system was noted in the preceding EJB2310I message for the VC ID.

**Module**
XNX25IPI
**EZB2317I**  VC vc call to address refused, cause=cause diagnostic=diagnostic

**Explanation**
An outgoing call on this virtual circuit to the indicated address was refused by the X.25 network, with the indicated cause and diagnostic bytes. Datagrams queued for the remote system are discarded. A new call is attempted when the TCP acknowledgment timer expires, or when a new connection is requested to the destination. TCP connections to the destinations handled by the remote system are unsuccessful if calls are not accepted in the initial connection time-out. See X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning for X.25 cause and diagnostic codes. See the X.25 network service provider for documentation about additional diagnostic codes.

**System action**
Processing continues.

**Operator response**
Verify that the remote system and the X.25 network are operational.

**System programmer response**
Assist the user as necessary.

**Module**
XNX25IPI

**Procedure name**
None.

---

**EZB2320I**  VC vc NPSI logon LU lu

**Explanation**
NPSI generated a VTAM session logon from the indicated LU for the virtual circuit. The NPSI LU logon is accepted.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Use the LU name for NPSI problem determination.

**Module**
XNX25IPI

**Procedure name**
None.
EZB2321E  VC vc session loss code

Explanation
X25IPI received the indicated virtual circuit (VC) status change code while the virtual circuit (VC) was not operational.

System action
The virtual circuit (VC) LU is enabled for automatic recovery by LOGAPPL or for manual restart.

Operator response
Notify the system programmer of the error.

System programmer response
Restart XNX25IPI if necessary.

Module
XNX25IPI

Procedure name
VCLOGON

EZB2322I  VC vc OPNDST complete

Explanation
A VTAM OPNDST request on a NPSI virtual circuit (VC) LU was posted complete.

System action
X25IPI continues processing.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCOPENC

EZB2323E  VC vc OPNDST did not complete

Explanation
A VTAM OPNDST request on a NPSI VC LU was posted complete, but the NIB is not marked open. The contents of the session data area (SDA) for this VC are dumped. The virtual circuit call is cleared.
**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Determine the state of the NPSI VC LU using the VTAM DISPLAY command.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2324I  VC vc LU lu ready**

**Explanation**
The virtual circuit LU session is ready for data transfer.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2325I  VC vc facilities: facilities**

**Explanation**
A list of facilities for this virtual circuit follows. The following are the facilities codes that can be noted:

**Codes**

* Description

- **pktpacket**
The noted packet size is used

- **precedence**
The noted DDN precedence is applied

- **priority**
Priority handling and charging is applied
revchg
Reverse charging is applied

standard
DDN standard service is used

wdwwindow
The noted window size is used
The noted facilities are applied to the connection.

System action
Processing continues.

Operator response
None.

System programmer response
Use the information to determine the X.25 network facilities being used on the connection.

Module
XNX25IPI

Procedure name
None.

EZB2326E VC vc facilities field unacceptable at offset offset

Explanation
X25IPI encountered an incorrect X.25 call facilities field from a remote system for this virtual circuit. The X.25 call facilities field is dumped, and the call is cleared.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the remote system that is generating the incorrect facilities. The X.25 address of the remote system was noted in the preceding EZB2301I or EZB2310I message for the VC ID.

Module
XNX25IPI

Procedure name
None.

EZB2330I VC vc call complete
Explanation
A VTAM OPNDST request on a NPSI virtual circuit (VC) LU was posted complete.

System action
X25IPI continues processing.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
MCHRDYIP

EZB2331I    VC vc data sent

Explanation
X25IPI has sent a datagram to the X.25 NCP Packet Switching Interface (NPSI).

System action
The contents of the X.25 data packet sequence are displayed in subsequent messages.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCLOGON

EZB2332I    VC vc data received

Explanation
X25IPI has received a data packet that contained an IP datagram.

System action
The contents of the X.25 data packet sequence are displayed in subsequent messages.
Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCRIPDT

EZW2333I  VC vc packet received

Explanation
X25IPI has received a General Access To X.25 Transport Extension (GATE) control packet on this virtual circuit (VC).

System action
The contents of the control packet are displayed in subsequent messages.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCRIPDT

EZW2334E  VC vc oversize data packet received, length=length

Explanation
A datagram was received on this virtual circuit that had a length exceeding the buffer size specified in the X25IPI configuration data set. Either the local or remote system is misconfigured. The packet is dumped, and the connection is cleared.

System action
Processing continues.

Operator response
Tell the system programmer about the error.
System programmer response
Check the buffer size on the Buffers record in the X25IPI configuration data set. The buffer size should be large enough to hold the maximum IP datagram permitted by the usage agreements of the X.25 network. The X.25 address of the remote system was noted in the preceding EZB2301I or EZB2310I message for the VC ID.

Module
XNX25IPI

Procedure name
None.

EZB2335E  VC vc unrecognized packet received

Explanation
X25IPI did not recognize the type code in a packet received from NPSI GATE. The unrecognized packet is dumped and discarded.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Determine the packet type from the dump; contact IBM software support services.

Module
XNX25IPI

Procedure name
None.

EZB2336I  VC vc inactivity timer expired

Explanation
The inactivity time for this virtual circuit passed with no data transferred. The virtual circuit connection is closed.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI
**Procedure name**
None.

**EZB2340I** VC vc call reset, cause=cause diagnostic=diagnostic

**Explanation**
A call reset was received for this virtual circuit with the indicated cause and diagnostic bytes. The reset is confirmed, and data transfer continues. See *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning* for a list of X.25 cause and diagnostic codes. See the provider of the X.25 network service for documentation that lists additional diagnostic codes.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2341I** VC vc reset collision

**Explanation**
A reset collision occurred on this virtual circuit. Data transfer resumes.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2342I** VC vc reset confirmed
Explanation
A reset on this virtual circuit was confirmed by the remote system. Data transfer resumes on the virtual circuit.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

EZB2343W  VC vc qualified data packet discarded

Explanation
A qualified data packet was received on an IP connection. Qualified data packets are not specified for use on IP connections. The qualified data packet is dumped and discarded.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Correct the remote system that is sending qualified data packets. The X.25 address of the remote system was noted in the preceding EZB2301I or EZB2310I message for the VC ID.

Module
XNX25IPI

Procedure name
None.

EZB2344W  VC vc interrupt indication received

Explanation
An interrupt indication was received for this virtual circuit. Interrupt packets are not specified for use on IP connections. The interrupt packet is dumped, and an interrupt response is sent.

System action
Processing continues.
**Operator response**
Tell the system programmer about the error.

**System programmer response**
Correct the remote system that is sending interrupt packets. The X.25 address of the remote system was noted in the preceding EZB2301I or EZB2310I message for the VC ID.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2345E  VC vc interrupt confirmed**

**Explanation**
An interrupt was confirmed for this virtual circuit. Interrupt packets are not specified for use on IP connections. The virtual circuit is reset, and data transfer continues.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Contact IBM software support services.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2346I  VC vc reset confirmation packet sent**

**Explanation**
A reset packet was received and X25IPI has sent a reset confirmation packet on this virtual circuit (VC).

**System action**
The contents of the X.25 reset confirmation packet are displayed in subsequent messages.

**Operator response**
None.

**System programmer response**
None.
Module
XNX25IPI

Procedure name
VCRIPDT

EZB2347I   VC vc interrupt confirm packet sent

Explanation
An interrupt packet was received and X25IPI has sent an interrupt confirmation packet on this virtual circuit (VC).

System action
The contents of the X.25 interrupt confirmation packet are displayed in subsequent messages.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCCONFI

EZB2348I   VC vc reset request packet sent

Explanation
X25IPI has sent a reset packet on this virtual circuit (VC) to reset the connection.

System action
The contents of the X.25 reset request packet are displayed in subsequent messages.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
VCCONFI

EZB2350I   VC vc call cleared, cause=cause diagnostic=diagnostic
**Explanation**

A call was cleared on this virtual circuit with the indicated cause and diagnostic bytes. A clear confirmation is sent, and the virtual circuit connection is closed. See *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning* for a list of X.25 cause and diagnostic codes. See the provider of the X.25 network service for documentation that lists additional diagnostic codes.

**System action**

Processing continues.

**Operator response**

None.

**System programmer response**

Use the cause and diagnostic codes to determine the reason the call was cleared.

**Module**

XNX25IPI

**Procedure name**

None.

**EZB2351I**  
VC vc connection terminated for address: sent count received count dropped count

**Explanation**

The number of datagrams sent, received, and dropped on the virtual circuit to the remote system with the indicated X.25 address is shown when the connection is ended.

**System action**

The VC LU session is closed.

**Operator response**

None.

**System programmer response**

None.

**Module**

XNX25IPI

**Procedure name**

None.

**EZB2352I**  
VC vc closed

**Explanation**

The call on the virtual circuit was ended. The virtual circuit is reused for new calls.
**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2353I VC vc clear request packet sent**

**Explanation**
X25IPI has sent a clear packet on this virtual circuit (VC) to clear the connection. XNX25IPI is in the process of closing the connection.

**System action**
The contents of the X.25 clear request packet are displayed in subsequent messages.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
VCCLROUT

**EZB2354I VC vc clear confirm packet sent**

**Explanation**
X25IPI has sent a clear confirmation packet on this virtual circuit (VC) for the clear request. XNX25IPI is in the process of closing the connection.

**System action**
The contents of the X.25 clear request packet are displayed in subsequent messages.

**Operator response**
None.
**EZB2355I**  VC vc close pending

**Explanation**
A close of this virtual circuit is partially completed.

**System action**
X25IPI waits for the remaining close events to occur.

**Operator response**
None.

**System programmer response**
None.

---

**EZB2356W**  VC vc unable to clear GATE call in state P2

**Explanation**
X25IPI needed to clear a NPSI GATE call while the call was pending. The NPSI GATE programming interface does not allow a call to be cleared in this state.

**System action**
X25IPI waits for action by the X.25 network.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Determine the reason the X.25 network or remote system did not respond to the call request.
**EZB2357E  VC vc clearing limit exceeded**

**Explanation**
No clear response was received from the remote system after four clear requests. The virtual circuit is marked as cleared.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Determine the reason the X.25 network or remote system did not respond to the clear request.

**Module**
XNX25IPI

---

**EZB2358I  VC vc clear confirmed**

**Explanation**
The remote system responded to a clear request on this virtual circuit. Virtual circuit termination continues; message EZB2352I should follow.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

---

**EZB2360I  VC vc closed LU**

**Explanation**
X25IPI has completed a VTAM CLSDST for a virtual circuit (VC) LU session.
**System action**  
The X.25 server continues processing.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
XNX25IPI

**Procedure name**  
VCCLOSRC

**EZB2361W VC vc clearing timer expired**

**Explanation**  
The remote system did not respond to a clear request in 180 seconds. The clear request is retried 4 times.

**System action**  
Processing continues.

**Operator response**  
Tell the system programmer about the error.

**System programmer response**  
Determine the reason the remote system did not respond to the clear request.

**Module**  
XNX25IPI

**Procedure name**  
None.

**EZB2362I VC vc clear request packet sent**

**Explanation**  
X25IPI has sent a clear packet on this virtual circuit (VC) to clear the connection. XNX25IPI is in the process of closing the connection.

**System action**  
The contents of the X.25 clear request packet are displayed in subsequent messages.

**Operator response**  
None.
System programmer response
None.

Module
XNX25IPI

Procedure name
VCCLRTMR

EZB2363E VC vc clearing limit exceeded

Explanation
No clear response was received from the remote system after four clear requests. The virtual circuit is marked as cleared.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Determine the reason the X.25 network or remote system did not respond to the clear request.

Module
XNX25IPI

Procedure name
None.

EZB2364I VC vc clear confirmation sent

Explanation
A clear request from the remote system was confirmed on this virtual circuit. Virtual circuit termination continues; message EZB2352I should follow.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI
**EJB236SI  VC vc clear sent**

**Explanation**
A clear request was sent on this virtual circuit. Virtual circuit termination continues; message EZB2352I should follow.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2366I  VC vc finished lu luname activity**

**Explanation**
X25IPI reports the completion of a pending VTAM request during virtual circuit (VC) closing.

**System action**
The X.25 server continues processing.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
VCCLWTRC

**EZB2367I  VC vc reuse delay ended**

**Explanation**
The timer expired for NPSI cleanup for this virtual circuit. A new call can now be made on this virtual circuit.
System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

EZB2368I VC vc retry busy call

Explanation
X25IPI tries a call again that previously received a busy indication from NPSI.

System action
Processing continues.

Operator response
If the error recurs, tell the system programmer about the error.

System programmer response
If the error recurs, check the number of virtual circuits specified on the Link record in the X25IPI configuration data set against the number defined in the NPSI configuration. Use the VTAM DISPLAY command to determine the state of the NPSI switched VC LUs.

Module
XNX25IPI

Procedure name
None.

EZB2370E VC vc inactive: VTAM request completion REQ=request

Explanation
A VTAM request has completed for this virtual circuit after the connection has ended.

System action
TCPIP continues.

Operator response
None.
**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2371E** VC vc inactive: session loss code code

**Explanation**
X25IPI received a session status notification with the indicated session loss code for this virtual circuit after the connection was ended. The notification is ignored. See message EZB2383I for the loss codes.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
None.

**EZB2372I** VC vc inactive: session cleanup

**Explanation**
Session cleanup is in progress for this inactive virtual circuit. The VC LU session is closed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI
EZE2373E  VC vc inactive: VTAM logon refused

Explanation
A late logon from a NPSI VC LU was refused after a call was cleared. The logon is rejected.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

EZE2374I  VC vc packet discarded on dead connection

Explanation
A packet was received on this virtual circuit after the connection was ended. The packet is discarded.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

EZE2381E  VC vc DATE error report command=command error=error

Explanation
X25IPI received a NPSI DATE error report for the indicated command and error. See X.25 Network Control Program Packet Switching Interface Host Programming for the error codes.
System action
X25IPI attempts to recover from the error.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

**EZB2382E** VC vc GATE error report command=command error=error

Explanation
X25IPI received a NPSI GATE error report for the indicated command and error. See X.25 Network Control Program Packet Switching Interface Host Programming for the error codes.

System action
X25IPI attempts to recover from the error.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

**EZB2383E** VC vc session loss code value on LU lu

Explanation
A session loss status notification was received for this virtual circuit with the indicated loss code.

Code
Description
Less than 50
VTAM LOSTERM exit codes
50
VTAM SCIP exit UNBIND
64000
VTAM NS exit CLEANUP
64001
VTAM NS exit session initiation failure

64002
VTAM NS exit session initiation negative response
The virtual circuit connection is closed.

System action
Processing continues.

Operator response
None.

System programmer response
Use the VTAM error code to determine why the session was unsuccessful.

Module
XNX25IPI

Procedure name
None.

EZB2384I   VC vc discarded packet packet in state state

Explanation
The indicated packet was discarded on this virtual circuit to accomplish error recovery. Virtual circuit error recovery is completed.

System action
Processing continues.

Operator response
See message EZB2021I VC for the call state codes.

System programmer response
Assist the user as necessary.

Module
XNX25IPI

Procedure name
None.

EZB2385E   VC vc received packet packet invalid in state state

Explanation
The indicated received packet was incorrect for the current virtual circuit state. The virtual circuit is reset or cleared.
System action
Processing continues.

Operator response
See message EZB2021I VC for the call state codes. Tell the system programmer about the error.

System programmer response
Determine the reason the X.25 network or remote system sent the incorrect packet.

Module
XNX25IPI

Procedure name
None.

EZB2386E  VC vc discarded packet packet in state state

Explanation
The indicated packet for VC was discarded because the state was not valid for the current virtual circuit state. The virtual circuit vc is cleared.

System action
Processing continues.

Operator response
None.

System programmer response
Determine the reason the remote X.25 network sent the incorrect state value for the circuit.

Module
XNX25IPI

Procedure name
None.

EZB2401E  VTAM GENCB failed, R15=value R0=value

Explanation
A VTAM GENCB call was unsuccessful with the indicated R15 and R0 values.

System action
X25IPI does not start.

Operator response
Tell the system programmer about the error.
System programmer response
Reassemble X25IPI with the most recent VTAM macro library.

Module
XNX25IPI

Procedure name
None.

EZB2402E VTAM ACB OPEN failed, R15=value ACBERFLG=value

Explanation
A VTAM OPEN request failed with the indicated R15 and ACBERFLG values.

System action
X25IPI does not start.

Operator response
Activate the X25IPI application in VTAM.

System programmer response
Use the VTAM error codes to determine why the VTAM ACB OPEN macro was unsuccessful. See z/OS Communications Server: SNA Programming where codes can be found.

Module
XNX25IPI

Procedure name
None.

EZB2403E VTAM SETLOGON failed, RTNCD=rc FDB2=value

Explanation
A VTAM SETLOGON request failed with the indicated return code and FDB2 values.

System action
Processing continues, but virtual circuit (VC) LU logons will fail.

Operator response
None.

System programmer response
Use the VTAM error code to determine the reason for the failure of the VTAM SETLOGON request.

Module
XNX25IPI
**Procedure name**

VTAMENAB

**EZB2404E** VTAM SETLOGON QUIESCE failed, RTNCD=rc FDB2=value

**Explanation**

A VTAM SETLOGON QUIESCE request failed with the indicated return code and FDB2 values.

**System action**

X25IPI termination continues.

**Operator response**

Tell the system programmer about the error.

**System programmer response**

Use the VTAM error codes to determine why the VTAM SETLOGON macro. See the z/OS Communications Server: SNA Programming where codes can be found.

**Module**

XNX25IPI

**Procedure name**

None.

**EZB2405E** VTAM ACB CLOSE failed, R15=value ACBERFLG=value

**Explanation**

A VTAM CLOSE request failed with the indicated return values.

**System action**

X25IPI termination continues.

**Operator response**

Tell the system programmer about the error.

**System programmer response**

Use the VTAM error codes to determine why the VTAM CLOSE macro was unsuccessful. See the z/OS Communications Server: SNA Programming where codes can be found.

**Module**

XNX25IPI

**Procedure name**

None.

**EZB2406E** VTAM GENCB failed, R15=value R0=value

**Explanation**

A VTAM GENCB call failed with the indicated R15 and R0 values.
**System action**
The X.25 server continues processing.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Reassemble X25IPI with the most recent VTAM macro library.

**Module**
XNX25IPI

**Procedure name**
VTAMSIN

---

**EZB2407E** VTAM VTAM request lu failed, RTNCD=rc FDB2=value

**Explanation**
The indicated VTAM request was unsuccessful with the indicated return code and FDB2 values. The virtual circuit can become unusable.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Use the VTAM error codes to determine why the VTAM request was unsuccessful. See the z/OS Communications Server: SNA Programming where codes can be found. Determine the state of the NPSI LU using the VTAM DISPLAY command. If all virtual circuits become unusable, HALT and restart the X25IPI application.

**Module**
XNX25IPI

**Procedure name**
None.

---

**EZB2410I** VTAM request complete for luname REQ=value

**Explanation**
X25IPI completed a VTAM request. See message EZB2411E for the request codes.

**System action**
The X.25 server continues processing.

**Operator response**
None.
System programmer response
None.

Module
XNX25IPI

Procedure name
VTAMCHK

EZB2411E  VTAM request failed for lu REQ=request RNTCD=rc FDB2=value sense=value (caller)

Explanation
The indicated VTAM request was unsuccessful with the indicated return code, FDB2, and sense values. The caller field specifies the routine that issued the request. The following are the request codes:

Request Code  Description
X'17'     OPNDST
X'1F'     CLSDST
X'22'     SEND
X'23'     RECEIVE

System action
If the error occurred on a NPSI MCH session, the MCH is shut down. If the error occurred on a VC session, the connection is closed.

Operator response
If the error occurred on a NPSI MCH session, activate the NPSI MCH in VTAM and use the X25IPI RESTART command to reacquire the MCH.

System programmer response
Use the VTAM error codes to determine why the VTAM ACB OPEN macro was unsuccessful. See the z/OS Communications Server: SNA Programming where codes can be found.

Module
XNX25IPI

Procedure name
None.

EZB2420I  Logon exit refusing session

Explanation
XNX25IPI is refusing a logon session because:

- For an MCH with LOGAPPL coded, the MCH is not defined by a LINK entry, or the MCH is not in a state where a logon is expected
• The logon was not initiated by X.25 NPSI
• A virtual circuit is not in a state where a logon is expected, possibly because a call request was unsuccessful. The NPSI LU session logon is refused.

System action
Processing continues.

Operator response
Notify the system programmer if the message is issued frequently, and connections are failing.

System programmer response
Check the source of the logons; determine why the call request was unsuccessful.

Module
XNX25IPI

Procedure name
None.

EZB2421E  Network services exit: Unrecognized request type

Explanation
XNX25IPI received an unrecognized Network Services RU from VTAM. The following are the supported RU:
• Cleanup Session (X'810629')
• Notify (X'810620')
• NS Procedure Error (X'010604')
VTAM is notified that the NS RU was not handled.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Obtain a VTAM buffer trace to determine the Network Services RU type.

Module
XNX25IPI

Procedure name
None.

EZB2422I  Network services exit: Cleanup session notification posted
Explanation
A VTAM Cleanup Session RU was received. The VTAM session is ended. A failure message for the MCH or virtual circuit (VC) session follows.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

EZB2423W  Network services exit: Session initiation failure notification posted

Explanation
A VTAM Session initiation failure RU was received. The VTAM session is ended. A failure message for the MCH or VC session can follow.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

EZB2424T  Session control exit: Unrecognized or unexpected request type

Explanation
An unexpected VTAM Session Control RU was received. ABEND code X'941' is displayed with this message.

System action
X25IPI abends.
Operator response
Tell the system programmer about the error.

System programmer response
Determine the source of VTAM session control request. Submit the ABEND dump to IBM software support services.

Module
XNX25IPI

Procedure name
None.

**EZB2425I**  Session control exit: UNBIND notification posted

**Explanation**
A VTAM Session Control RU of type UNBIND was received. The VTAM session is ended. A failure message for the MCH or VC session can follow.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

Module
XNX25IPI

Procedure name
None.

**EZB2431E**  VTAM terminating, reason unknown

**Explanation**
A VTAM Shutdown notification was received. The cause is unknown. This rules out a standard shutdown, HALT QUICK, INACT, or HALT CANCEL reason codes.

**System action**
X25IPI ends.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Determine the source of the Shutdown notification.
Module
NX25IPI

Procedure name
None.

EZB2432I  VTAM HALT issued, drain flag set

Explanation
A VTAM HALT request was issued.

System action
X25IPI ends after current connections are closed. New connections are refused.

Operator response
Restart X25IPI, if appropriate, after VTAM resumes.

System programmer response
Assist the user as necessary.

Module
XNX25IPI

Procedure name
None.

EZB2433I  VTAM HALT QUICK or VARY INACT, application issued

Explanation
A VTAM HALT QUICK or VARY INACT request was issued.

System action
X25IPI ends.

Operator response
Restart X25IPI, if appropriate, after VTAM resumes.

System programmer response
Assist the user as necessary.

Module
XNX25IPI

Procedure name
None.

EZB2434I  VTAM HALT CANCEL issued
A VTAM HALT CANCEL request was issued.

X25IPI ends.

Restart X25IPI, if appropriate, after VTAM resumes.

Assist the user as necessary.

A DLC connection from the TCPIP address space was refused because a VTAM HALT command had been issued.

Processing continues.

HALT and restart the X25IPI application when VTAM is restarted.

Assist the user as necessary.

A DLC connection from the TCPIP address space was refused because a VTAM HALT command had been issued.

Processing continues.

HALT and restart the X25IPI application when VTAM is restarted.

Assist the user as necessary.

An IP datagram was received from TCPIP for transmission to the indicated address, which is not listed in the Dest entries in the X25IPI configuration data set. The IP datagram is discarded.

Processing continues.
Operator response
Tell the system programmer about the error.

System programmer response
Add a Dest record to the X25IPI configuration data set for the IP address or network. Check the BEGINROUTES entries in hlq.PROFILE.TCPIP for a misrouted network number.

Module
XNX25IPI

Procedure name
None.

**EZB2462E**    IP AS_userid rejected message: too long IP AS_userid rejected message: too short

Explanation
A DLC message was received from the TCPIP address space which was too long or too short. The message is rejected.

System action
Processing continues.

Operator response
Tell the system programmer about the error.

System programmer response
Contact IBM software support services.

Module
XNX25IPI

Procedure name
None.

**EZB2480I**    IP AS_id disconnected: sent count received count dropped count

Explanation
The number of datagrams sent, received, and dropped on the DLC connection to the TCPIP address space are shown when the connection is terminated.

System action
X25IPI ends.

Operator response
None.

System programmer response
None.
Module
XNX25IPI

Procedure name
None.

EZB2491I DLC AS asname path severed

Explanation
The DLC connection to TCPIP has been severed, most likely because TCPIP is being terminated or the X25NPSI device has been STOPped.

System action
X25IPI exits.

Operator response
Restart X25IPI, if appropriate, after TCPIP is restarted.

System programmer response
None.

Module
XNX25IPI

Procedure name
IPRDYPN

EZB2492I DLC AS_path accepted for userid userid

Explanation
A DLC connection from the TCPIP address space was accepted. IP datagrams are transferred.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
None.

EZB2493I DLC received data
**Explanation**
X25IPI displays the contents of an IP datagram received from TCPIP in subsequent messages.

**System action**
The X.25 server continues processing.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
IPRDYPN

---

**EZB2494I** DLC send datagram length length next offset offset

**Explanation**
X25IPI displays the length of the datagram to be sent to TCPIP.

**System action**
The X.25 server continues processing.

**Operator response**
None.

**System programmer response**
None.

**Module**
XNX25IPI

**Procedure name**
IPSENDDG

---

**EZB2495I** DLC send total length length

**Explanation**
X25IPI displays the total length of IP datagrams to be sent to TCPIP.

**System action**
The X.25 server continues processing.
Operator response
None.

System programmer response
None.

Module
XNX25IPI

Procedure name
IPSENDDG

EJB2496E XNX25IUT failed: CSM storage problem

Explanation
XNX25IUT interface returned a return code of 4. This indicates an error with CSM storage.

System action
X25IPI closes connection with TCPIP.

Operator response
Tell the system programmer about the error.

System programmer response
Check for correct installation of the MVS TCPIP product.

Module
XNX25IPI

Procedure name
None.

EJB2497E DLC function failed: STAFC= reason

Explanation
XNX25IUT interface returned a return code of 8. The indicated DLC function, issued by XNX25IPI, failed for the reason indicated in the STAFC reason field.

1 INIT
2 SEND
3 RECV
4 CLEAR
5 TERM
**System action**
The DLC connection is severed.

**Operator response**
Tell the system programmer.

**System programmer response**
Perform the action described in *z/OS Communications Server: IP and SNA Codes* for the indicated status code.

**Module**
XNX25IPI

**Procedure name**
None.

---

**EZB2498E**  XNX25IUT failed: Unexpected return code

**Explanation**
XNX25IUT interface returned a return code other than 0, 4, or 8.

**System action**
Processing continues.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Contact IBM software support services.

**Module**
XNX25IPI

**Procedure name**
None.
Chapter 5. EZB3xxxx messages

**EZB3000I*** Can't find initialize address for server *server: return_code*

**Explanation**
The local host was unable to find the address of the Domain Name Server. Without the Domain Name Server, the local host is unable to determine the addresses of other hosts on the network.

**System action**
TCPIP continues.

**Operator response**
Notify the system programmer of the problem.

**System programmer response**
Add the address of the Domain Name Server (NSINTERADDR) to the *hlq.TCPIP.DATA* data set. For more information about the *hlq.TCPIP.DATA* data set, see z/OS Communications Server: IP Configuration Reference for more information.

**Module**
NSLOOKUP

**Procedure name**
main

---

**EZB3001I**  Usage:  

- `nslookup [-opt ...] # interactive mode using default server`
- `nslookup [-opt ...] - server #interactive mode using default server`
- `nslookup [-opt ...] host #just look up ‘host’ using default server`
- `nslookup [-opt ...] host #just look up ‘host’ using ‘server’`

**Explanation**
These messages give the format and usage for the NSLOOKUP command. For more information about the NSLOOKUP command, see the z/OS Communications Server: IP User's Guide and Commands.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.
Module
NSLOOKUP

Procedure name
Usage

EZB3006I  SetDefaultServer: invalid name: name

Explanation
The name specified for the default Domain Name Server does not correspond to a local host. The Domain Name Server is not initialized.

System action
TCPIP continues.

Operator response
Notify the system programmer of the problem.

System programmer response
Correct the default Domain Name Server name (NSINTERADDR) in the hlq.TCPIP.DATA data set. For more information about configuring the Domain Name Server, see z/OS Communications Server: IP Configuration Reference.

Module
NSLOOKUP

Procedure name
SetDefaultServer

EZB3007I  *** Can't find address for server server: reason

Explanation
TCPIP was unable to change to another Domain Name Server because it could not find an address for the new Domain Name Server. The reason it could not find the address is displayed in the message.

System action
TCPIP continues.

Operator response
Notify the system programmer of the problem.

System programmer response
Respond as indicated by the reason portion of this message.

Module
NSLOOKUP
**Procedure name**
SetDefaultServer

**EZB3008I***  **No query_type (query_abbrev.) records available for host**

**Explanation**
No information of the type requested is available for the indicated host.

**System action**
TCPIP continues.

**Operator response**
Send a PING to the indicated host to determine if it is reachable through the network. Check the SEZAINST data set to make sure that recursion has been requested. Recursion will allow the Domain Name Server for the local zone to communicate with other Domain Name Servers to determine the requested address. If the error persists, notify the system programmer. For more information about recursion, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
Make sure that the Domain Name Server is started and online. If necessary, update the address tables for the Domain Name Server to include the indicated host.

**Module**
NSLOOKUP

**Procedure name**
DoLookup

**EZB3009I***  **Request to server timed-out**

**Explanation**
A request to the indicated server reached the end of its time to live. The request is not answered.

**System action**
TCPIP continues.

**Operator response**
Resubmit the request. If the error persists, notify the system programmer.

**System programmer response**
Check the indicated server to determine why it is not answering requests.

**Module**
NSLOOKUP

**Procedure name**
DoLookup

**EZB3010I***  **Server can't find host: reason**
**Explanation**
The Domain Name Server is unable to find an address for the indicated host. The reason is displayed in the message.

**System action**
TCPIP continues.

**Operator response**
Notify the system programmer of the problem.

**System programmer response**
Respond as indicated by the reason portion of the message.

**Module**
NSLOOKUP

**Procedure name**
DoLookup

---

**EZB3011E  *** Can't open file for writing**

**Explanation**
The Domain Name Server is unable to open a file to write the answer to a request. The request is not answered.

**System action**
TCPIP continues.

**Operator response**
Notify the system programmer of the error.

**System programmer response**
Make sure that the Domain Name Server has the correct write authority.

**Module**
NSLOOKUP

**Procedure name**
LookupHost, LookupHostWithServer

---

**EZB3012I  > request**

**Explanation**
This message is written to the output file. It echoes the request to the Domain Name Server that produced the output file.

**System action**
TCPIP continues.
Operator response
None.

System programmer response
None.

Module
NSLOOKUP

Procedure name
LookupHost, LookupHostWithServer

EZB3013W  *** Can't find address for server server: reason

Explanation
The default Domain Name Server was unable to get information about the requested Domain Name Server. The reason for the error is displayed in the message. The request is not answered.

System action
TCPIP continues.

Operator response
Notify the system programmer of the error.

System programmer response
Respond as indicated by the reason portion of this message.

Module
NSLOOKUP

Procedure name
LookupHostWithServer

EZB3014E  *** Invalid set command

Explanation
The SET subcommand, which is used to set or change the options for the NSLOOKUP command, was submitted with no associated option. The SET subcommand is not accepted.

System action
TCPIP continues.

Operator response
If the NSLOOKUP command was submitted in interactive mode, resubmit the command specifying a valid option after the SET subcommand. To accept the current options to the NSLOOKUP command, omit the SET subcommand. If the NSLOOKUP command was submitted from the user_id.NSLOOKUP.ENV data set, edit the data set, including a valid option after the SET subcommand to change the NSLOOKUP options, or deleting the SET subcommand to accept the current option settings. For more information about the SET subcommand, the
options to the NSLOOKUP command, and the user_id.NSLOOKUP.ENV data set, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
Assist the user as necessary.

**Module**
NSLOOKUP

**Procedure name**
SetOption

**EZB3015I d2 mode disabled; still in debug mode**

**Explanation**
The NO D2 option has been accepted, disabling the high-level tracing for the Domain Name Server. The server is still in debug mode.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NSLOOKUP

**Procedure name**
SetOption

**EZB3016E invalid port value: port**

**Explanation**
The PORT option to the SET subcommand, which specifies the port to use when contacting the Domain Name Server, was submitted with an incorrect value for the port. The given value was either out of the range of valid ports, or the value of a port that is already assigned. The PORT option is not accepted.

**System action**
TCPIP continues.

**Operator response**
Resubmit the NSLOOK command, specifying a valid port in the PORT option to the SET subcommand. The Domain Name Server is a well-known service, and is allocated port 53. For more information about the NSLOOKUP command, its subcommands and options, see the z/OS Communications Server: IP User's Guide and Commands.
**System programmer response**
Assist the user as necessary.

**Module**
NSLOOKUP

**Procedure name**
SetOption

**EZB3017E invalid retry value: value**

**Explanation**
The RETRY option to the SET subcommand of the NSLOOKUP command was submitted with an incorrect value. The NSLOOKUP command is not accepted. The RETRY option specifies the number of times a request is resent. If the RETRY option is set to 0, no requests are sent, resulting in the error message "no response from server".

**System action**
TCPIP continues.

**Operator response**
Resubmit the NSLOOKUP command with a valid value for the RETRY option of the SET subcommand. For more information about the NSLOOKUP command and its subcommands and options, see the z/OS Communications Server: IP User’s Guide and Commands.

**System programmer response**
Assist the user as necessary.

**Module**
NSLOOKUP

**Procedure name**
SetOption

**EZB3018E invalid timeout value: value**

**Explanation**
The TIMEOUT option to the SET subcommand of the NSLOOKUP command was submitted with an incorrect value. The NSLOOKUP command is not accepted. The TIMEOUT option specifies the number of seconds to wait before canceling a request. If the TIMEOUT option is set to 0, requests are canceled immediately and are never answered.

**System action**
TCPIP continues.

**Operator response**
Resubmit the NSLOOKUP command with a valid value for the RETRY option of the SET subcommand. For more information about the NSLOOKUP command and its subcommands and options, see the z/OS Communications Server: IP User’s Guide and Commands.
System programmer response
Assist the user as necessary.

Module
NSLOOKUP

Procedure name
SetOption

EZB3019E *** Invalid option: option

Explanation
The SET subcommand was submitted with an option that the Domain Name Server did not recognize. The command is not accepted.

System action
TCPIP continues.

Operator response
Check the spelling and syntax and resubmit the command. For more information about valid options for the SET subcommand, see the z/OS Communications Server: IP User's Guide and Commands.

System programmer response
Assist the user as necessary.

Module
NSLOOKUP

Procedure name
SetOption

EZB3020I Set options:
EZB3021I nodebug nODEFname nosearch norecurse
EZB3025I nod2 noclgnoretc port=port
EZB3029I querytype=type class=class timeout=number retry=number
EZB3033I root=server
EZB3034I domain=domain
EZB3035I nobrackets
EZB3194I diff/time/hostname
EZB3036I srchlist=domain1|domain2|domain3

Explanation
These messages display the state information used by the resolver library and other options set by the user. The prefix no to an option name indicates that the option is not selected. For more information about these options, see the z/OS Communications Server: IP User's Guide and Commands.
System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
NSLOOKUP

Procedure name
ShowOptions

EZB3038E  *** Can't initialize resolver.

Explanation
The Domain Name Server was unable to initialize the resolver library routines, which are used by clients to request resolution by the Domain Name Server. This indicates that there was an error in specifying the configuration file for the resolver, the RESOLVER address space was not started, or the Domain Name Server was unable to allocate the configuration data set. See z/OS Communications Server: IP Configuration Reference for an explanation of the search order for the configuration data set.

System action
Initialization halts. TCPIP continues.

Operator response
Check the syntax of the configuration data set, start the RESOLVER address space, and resubmit the command. If the error persists, specify the configuration options using the NSLOOKUP command and its parameters. For more information about the configuration data set, see z/OS Communications Server: IP Configuration Reference. For more information about the NSLOOKUP command and its parameters, see the z/OS Communications Server: IP User's Guide and Commands.

System programmer response
Make more storage available to the Domain Name Server if necessary.

Module
NSLOOKUP

Procedure name
Main

EZB3039E  *** Can't find server address for ‘server’:

Explanation
While initialing the resolver library used by the clients to request domain name resolution, the Domain Name Server was unable to find an IP address for the indicated server.
**System action**
Initialization halts. TCPIP continues.

**Operator response**
Correct the configuration data set to include an IP address for the default Domain Name Server. For more information about the domain name server configuration data set, see z/OS Communications Server: IP Configuration Reference.

**System programmer response**
Assist the user as necessary.

**Module**
NSLOOKUP

**Procedure name**
ReadRC

---

**EZB3040E  *** Can't find server name for address address: reason**

**Explanation**
While initializing the resolver library, which is used by the client to request domain name resolution, the Domain Name Server was unable to find a server name for the indicated address.

**System action**
Initialization halts. TCPIP continues.

**Operator response**
Correct the TCPIP.DATA data set to include a NSINTERADDR statement for the IP address of the Domain Name Server, or enter the NSLOOKUP command, using the *server_name* and *server_address* parameters to specify the default Domain Name Server. For more information about the Domain Name Server configuration and the steps involved, see z/OS Communications Server: IP Configuration Reference. For more information about the NSLOOKUP command and its parameters, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
Assist the user as necessary.

**Module**
NSLOOKUP

**Procedure name**
Usage

---

**EZB3041E  *** Default servers are not available**

**Explanation**
The Domain Name Servers specified as the default servers in the resolver library used by the client to request address resolution by the Domain Name Server are not available.
**System action**
Initialization halts. TCPIP continues.

**Operator response**
Use the NSLOOKUP command with the server_name and server_address parameters to specify a different default server. If the error persists, notify the system programmer.

**System programmer response**
Make sure that the indicated Domain Name Servers are started and online.

**Module**
NSLOOKUP

**Procedure name**
Usage

**EZB3042I > command**

**Explanation**
This message is the command prompt for interactive NSLOOKUP sessions. For more information about interactive sessions using NSLOOKUP, see the z/OS Communications Server: IP User's Guide and Commands.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NSLOOKUP

**Procedure name**
Usage

**EZB3071W (name truncated?)**

**Explanation**
This message indicates a compression error in the resource records generated by a name server query.

**System action**
TCPIP continues.

**Operator response**
Notify the system programmer.
System programmer response
Check the indicated Domain Name Server to determine the cause of the compression error. For more information about resource records and the Domain Name Server, see z/OS Communications Server: IP Configuration Reference.

Module
BD@DEBUG

Procedure name
Print_rr

**EJB3078I**

**address, class = number, len = length**

**Explanation**
This message is sent to the trace file if the D2 option was specified for the NSLOOKUP command. It displays the class and length of an address type resource record generated by a name server query.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

Module
BD@DEBUG

Procedure name
Print_rr

**EJB3089I**

**origin = name**

**EJB3090I**

**mail addr = name**

**EJB3091I**

**serial = serial_number (name)**

**EJB3092I**

**refresh = refresh_time (time)**

**EJB3093I**

**retry = retry_time (time)**

**EJB3094I**

**expire = time_to_live (time)**

**EJB3095I**

**minimum ttl = min_time_to_live (time)**

**Explanation**
These messages are sent to the trace file if the DEBUG option was specified. They give information about the options selected for the Domain Name Server. For more information about the Domain Name Server, see the z/OS Communications Server: IP User's Guide and Commands.

**System action**
TCPIP continues.
Operator response
None.

System programmer response
None.

Module
BD@DEBUG

Procedure name
Print_rr

**EZB3097E**   errors = name

Explanation
This message indicates the host to which errors will be sent if they are encountered by the Domain Name Server.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@DEBUG

Procedure name
Print_rr

**EZB3105I**   NULL (dlen length)

Explanation
This message is sent to the trace file if the DEBUG option was specified. It indicates that a resource record generated by a name server request has a type of null and the indicated length.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
**Module**
BD@DEBUG

**Procedure name**
Print_rr

---

**EZB3106E**  ??? unknown type type ???

**Explanation**
This message is sent to the trace file if the DEBUG option was specified. It indicates that a resource record generated by a name server request has a type that the Domain Name Server does not recognize. The type is displayed in the message.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
Check the indicated Domain Name Server to determine why it is generating incorrect resource records.

---

**EZB3108E**  *** Error: record size incorrect (actual_record_size != stated_record_size)***

**Explanation**
This message is sent to the trace file if the DEBUG option was specified. It indicates that the Domain Name Server received a record that is not equal to the record size set in the hlq.PROFILE.TCPIP data set.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
Change the record sizes in the hlq.PROFILE.TCPIP data set so that the stated record size matches the actual record size.

---

**Module**
BD@DEBUG
**Procedure name**
Print_rr

**EZB3110I  Non-authoritative answer:**

**Explanation**
The name for which the client requested resolution is outside the current Domain Name Server's zone of authority, and recursive resolution was not requested. The Domain Name Server returns the name of the name server most likely to be able to resolve the request. The client issues a request to this name server.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
BD@GET

**Procedure name**
FreeHostInfoPtr

**EZB3112I  Authoritative answers can be found from: server(s)**

**Explanation**
The addresses requested by the client are outside the current Domain name server's zone of authority, and no recursive resolution was requested. The current Domain Name Server returns the names of the servers most likely to have information about the names being requested.

**System action**
TCPIP continues.

**Operator response**
Query the indicated name servers for more information.

**System programmer response**
Assist the user as necessary.

**Module**
BD@GET

**Procedure name**
FreeHostInfoPointer

**EZB3113I  Aliased to host**
**Explanation**
The address requested by the client is not in the zone of authority for the current Domain Name Server. The name server returns the name, or alias, of the name server most likely to have the address in its domain.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
BD@GET

**Procedure name**
GetHostByName

**EZB3116E  *** ls: invalid request request**

**Explanation**
The Domain Name Server received a request it did not recognize. The Domain Name Server is unable to process this request. The request is displayed in this message.

**System action**
TCPIP continues.

**Operator response**
Check the syntax and resubmit the request.

**System programmer response**
Assist the user as necessary.

**Module**
BD@LIST

**Procedure name**
ListHostsByType

**EZB3117I  *** Can't list domain domain: reason**

**Explanation**
This message is sent to the trace file. The procedure ListHostsByType, which lists the hosts known to the Domain Name Server, was unsuccessful for the indicated reason.

**System action**
TCPIP continues.
Operator response
None.

System programmer response
Respond as indicated by the reason portion of this message.

Module
BD@LIST

Procedure name
ListHostsByType

EZB3119I [host]

Explanation
This message displays the name of the local host.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
ListHostsByType

EZB3120E *** Can't open file for writing

Explanation
The function OpenFile, which parses the command string for a file name and opens the file for writing, was unsuccessful, indicating an incorrect command string or an error opening the file. The data set is not opened.

System action
TCPIP continues.

Operator response
Check the syntax and resubmit the command. If the error persists, notify the system programmer.

System programmer response
Make sure that the user has the correct authority to write the indicated data set.
Module
BD@LIST

Procedure name
ListHosts

EZB3121I  >command

Explanation
This message is written to a file opened by the function OpenFile. It echoes the last command submitted by the user.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
ListHosts

EZB3122I  Alias

Explanation
This message precedes the alias of a host. Hosts can be addressed by their alias, their network address, or their canonical name. For more information about aliases, see the z/OS Communications Server: IP User's Guide and Commands.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
ListHosts
EZB3124I  Received *number* records.

Explanation
The Domain Name Server received the indicated number of records in response to a query.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
ListHosts

EZB3125E    *** ls: error receiving zone transfer:

EZB3126I    result: *result*, answers = *number*, authority = *number*, additional = *number*

Explanation
The client was unable to read the response from an LS command, used to list the name servers in other domains known to the local Domain Name Server. The data from the LS command is not transferred.

System action
TCPIP continues.

Operator response
Notify the system programmer of the error.

System programmer response
Check the Domain Name Server to make sure that it is correctly configured to be compatible with the other Domain Name Servers on the system. For more information about configuring the Domain Name Server, see z/OS Communications Server: IP Configuration Reference.

Module
BD@LIST

Procedure name
ListHosts

EZB3127I    *domain_name*

Explanation
This message displays the domain name for which an answer returned from the name server is valid.
System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
PrintListInfo

EZB3131I  (address_protocol_identifiers)

Explanation
This message indicates the IP address protocol being used by NSLOOKUP. For more information about IP address protocols, see the z/OS Communications Server: IP User's Guide and Commands.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
PrintListInfo

EZB3132I  (dlen = length?)

Explanation
This message indicates the length of data in a response received by NSLOOKUP.

System action
TCPIP continues.

Operator response
None.
System programmer response
None.

Module
BD@LIST

Procedure name
PrintListInfo

EZB3133I  ***

Explanation
This message precedes other informational messages.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
PrintListInfo

EZB3134I  server, type class

Explanation
This message displays the name of the server being queried, the type of query, and the class of address being requested.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@DEBUG
**Procedure name**
Print_cdname_sub

**EZB3143I**  *text data*

**Explanation**
This is variable data which is displayed in response to the NSLOOKUP command. It is defined by the installation in the name server data base.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
BD@DEBUG

**Procedure name**
Print_rr

**EZB3144I**  *number*

**Explanation**
This message displays the user or group ID from a name server query.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
BD@LIST

**Procedure name**
PrintListInfo

**EZB3145I**  *number*

**Explanation**
This message displays the protocol value taken from a WKS resource record. For more information about resource records, see the z/OS Communications Server: IP User's Guide and Commands.

298  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
PrintListInfo

EZB3146  protocol

Explanation
This message displays the mnemonic for the protocol value taken from a well known services (WKS) resource record. For more information about resource records, see the z/OS Communications Server: IP User's Guide and Commands.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
PrintListInfo

EZB3150E  *** Can't open data_set for reading

Explanation
The Domain Name Server was unable to read from the indicated data set. The Domain Name Server query is not answered.

System action
TCPIP continues.

Operator response
Make sure the indicated data set is in storage accessible to the Domain Name Server.
System programmer response
Assist the user as necessary.

Module
BD@LIST

Procedure name
ViewList

EZB3151I  *host*

Explanation
This message displays the output of the LS function.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@LIST

Procedure name
ViewList

EZB3154I  Finger: no current host defined.

Explanation
No host has been defined to the current finger server, which displays information about users of a remote host. The FINGER query is not answered.

System action
TCPIP continues.

Operator response
Resubmit the FINGER command, using the format FINGER@host to define the remote host to be used to complete the request.

System programmer response
Assist the user as necessary.
**Procedure name**
Finger

**EZB3155I**  
Finger: unknown service

**Explanation**
The finger server, which gives information about the users of a remote host, received a request for a service that it does not recognize. The request is not processed.

**System action**
TCPIP continues.

**Operator response**
Check the syntax and resubmit the command. For more information about the finger server, see the z/OS Communications Server: IP User’s Guide and Commands.

**System programmer response**
Assist the user as necessary.

**Module**
BD@LIST

**Procedure name**
Finger

**EZB3156I**  
> data_set_name

**Explanation**
This message displays the name of the data set to which information returned by the finger server will be sent.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
BD@LIST

**Procedure name**
Finger

**EZB3169E**  
*** Can't allocate memory
Explanation
The Domain Name Server was unable to allocate storage to hold lookup tables because no data set name was specified.

System action
TCPIP continues.

Operator response
Check the syntax and resubmit the command. If the error persists, notify the system programmer.

System programmer response
Make sure that sufficient storage is available for the tables needed by the Domain Name Server. If the error persists, check for allocation errors on the server.

Module
BD@SUBR

Procedure name
Malloc

EJB3170I Server: server
EJB3171I Addresses: addresses
EJB3172I Address: address

Explanation
These messages give the name and addresses of the server being queried.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@SUBR

Procedure name
PrintHostInfo

EJB3175I Aliases: alias_names

Explanation
This message is displayed with message EJB3170I and gives the alias names corresponding to the indicated addresses.
System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@SUBR

Procedure name
PrintHostInfo

EZB3176I  Served by:
EZB3177I    servers
EZB3178I    , internet_address

Explanation
These messages are displayed with message EZB3170I. They list the name servers for the indicated addresses.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@SUBR

Procedure name
PrintHostInfo

EZB3181W  unknown query class: class

Explanation
The Domain Name Server received a query of a class it does not recognize. Repeated queries of this class will tie up the name server, preventing it from replying to valid queries. The query is not processed.

System action
TCPIP continues.
**Operator response**

Notify the system programmer of the problem.

**System programmer response**

Check the client to determine why it is sending incorrect queries. Take the client offline until the problem is solved.

**Module**

BD@SUBR

**Procedure name**

StringToClass

---

**EZB3182W**  unknown query type: **type**

**Explanation**

The Domain Name Server received a query of a type it does not recognize. The query cannot be processed, and too many queries of this type will tie up the name server, preventing it from replying to valid queries. The query is not processed.

**System action**

TCPIP continues.

**Operator response**

Notify the system programmer of the problem.

**System programmer response**

Check the indicated client to determine why it is sending queries of an incorrect type. Take the client offline until the problem is corrected.

**Module**

BD@SUBR

**Procedure name**

StringToType

---

**EZB3194I**  **type**stamp

**Explanation**

This message indicates the type of time stamping selected for the NSLOOKUP command. Possible values are:

**Value**

<table>
<thead>
<tr>
<th>Indicates</th>
<th>time</th>
<th>The time is displayed before each output line.</th>
</tr>
</thead>
<tbody>
<tr>
<td>diff</td>
<td>The time is displayed before each output line only when the time changes.</td>
<td></td>
</tr>
<tr>
<td>no</td>
<td>No time stamping is selected.</td>
<td></td>
</tr>
</tbody>
</table>
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NSLOOKUP

**Procedure name**
ShowOptions

**EZB3195I [host]**

**Explanation**
This message displays the name of the local host.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
BD@LIST

**Procedure name**
ListHostsByType

**EZB3196W no DOMAIN is entered, using default**

**Explanation**
The Domain Name Server received a command that does not specify a domain. The name server uses the default domain.

**System action**
TCPIP continues.

**Operator response**
Reenter the command, specifying the correct domain if necessary.
System programmer response
Assist the user as necessary.

Module
NSLOOKUP

Procedure name
SetOptions

EZB3197W   no ROOT is entered, using default

Explanation
The Domain Name Server received a command that did not specify the root name server to be used. The Domain Name Server uses the default server.

System action
TCPIP continues.

Operator response
Resubmit the command, specifying the root server if necessary.

System programmer response
Assist the user as necessary.

Module
NSLOOKUP

Procedure name
SetOptions

EZB3198W   no SRCHLIST is entered, using default

Explanation
The Domain Name Server received a command that did not specify the search list to be used. The Domain Name Server uses the default search list.

System action
TCPIP continues.

Operator response
Resubmit the command, specifying the search list if necessary.

System programmer response
Assist the user as necessary.
Procedure name
SetOptions

EZB3201I  text = text

Explanation
This message displays the contents of a text type query.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
BD@DEBUG

Procedure name
Print_rr

EZB3202E  *** Syntax error in option option.

Explanation
The indicated option of the NSLOOKUP command was submitted with incorrect syntax. The option is ignored.

System action
TCPIP continues.

Operator response
Resubmit the NSLOOKUP command with the correct syntax for all options. For more information about the NSLOOKUP command and the correct syntax for its options, see the z/OS Communications Server: IP User's Guide and Commands.

System programmer response
Assist the user as necessary.

Module
NSLOOKUP

Procedure name
comml_option

EZB3203E  *** The '=' sign is missing in option option

Explanation
The indicated option of the NSLOOKUP command was submitted without a '='. The option cannot be processed.
**System action**
TCPIP continues.

**Operator response**
Resubmit the NSLOOKUP command with the correct syntax for all options. For more information about the NSLOOKUP command and its options, see the z/OS Communications Server: IP User's Guide and Commands.

**System programmer response**
Assist the user as necessary.

**Module**
NSLOOKUP

**Procedure name**
comml_option

---

**EZB3205I**  
**local host**

**Explanation**
This is the name of the local host or the name of a server that knows about the local domain.

**System action**
NSLOOKUP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
BD@LIST

**Procedure name**
ListSubr

---

**EZB3206I**  
nslookup [ -option ... ] [ host-to-find | - [ server ]]  

**EZB3207I**  
Commands: (identifiers are shown in uppercase, <> means optional)

**EZB3208I**  
NAME - print info about the host\domain NAME using default server

**EZB3209I**  
NAME1 NAME2 - as above, but use NAME2 as server

**EZB3210I**  
help or ? - print info on common commands;

**EZB3211I**  
use TSO's help nslookup for more details

**EZB3212I**  
set OPTION - set an option

**EZB3213I**  
<no>debug - print debugging information
EZB3216I  <no>d2 - print exhaustive debugging information
EZB3217I  <no>defname - append domain name to each query
EZB3218I  <no>recurse - ask for recursive answer to each query
EZB3219I  <no>vc - always use a virtual circuit
EZB3220I  domain=NAME - set default domain name to NAME
EZB3221I  srchlist=N1<⁄N2⁄...⁄N6> - set domain to N1 and search
EZB3222I  list to N1,N2, etc.
EZB3223I  root=NAME - set root server to NAME
EZB3224I  retry=X - set number of retries to X
EZB3225I  timeout=X - set initial time-out interval to X seconds
EZB3226I  querytype=X - set query type, e.g.,
EZB3227I  A,ANY,CNAME,HINFO,MC,NS,PTR,SOA,WKS
EZB3228I  type=X - synonym for querytype
EZB3229I  class=X - set query class to one of IN (Internet),
EZB3230I  CHAOS, HESIOD or ANY
EZB3231I  server NAME - set default server to NAME, using current
EZB3232I  default server
EZB3233I  lsrvr NAME - set default server to NAME, using initial server
EZB3234I  finger <USER> - finger the optional NAME at the current
EZB3235I  default host
EZB3236I  root - set current default server to the root
EZB3237I  ls <opt> DOMAIN [> DATASET] - list addresses in DOMAIN
EZB3238I  (optional: output to DATASET)
EZB3239I  -a - list canonical names and aliases
EZB3240I  -h - list HINFO (CPU type and operating system)
EZB3241I  -s - list well-known services
EZB3242I  -d - list all records
EZB3243I  -t TYPE - list records of the given type
EZB3244I  (e.g., A, CNAME, MX, etc.)
EZB3245I  view DATASET - sort an ‘ls’ output file and view it with more
EZB3246I  exit - exit the program

Explanation
This is the summary of available commands and options that you can review by typing help or ? at the > prompt (see message EZB3042I), once you have entered the NSLOOKUP interactive session. For details about any of these commands or options, see z/OS Communications Server: IP User’s Guide and Commands.

System action
The name server continues until you type exit at the > prompt (see message EZB3042I).
Operator response
Use these help messages as a guide for entering subsequent NSLOOKUP commands.

System programmer response
None.

Module
NSLOOKUP

Procedure name
PrintHelp()

EZB3250I invalid dig option option

Explanation
The option indicated was entered at the command line and is not a valid DIG option.

System action
TCPIP continues.

Operator response
Enter a valid DIG option. z/OS Communications Server: IP User’s Guide and Commands contains the DIG command syntax and the valid options.

System programmer response
None.

Module
DIG

Procedure name
PrintHelp

EZB3251I no dig -T value specified

Explanation
No wait time (-T value) was specified. Wait time is the time to wait between successive queries when operating in batch mode. The default wait time is 0, which indicates no wait time.

System action
TCPIP continues.

Operator response
If you wish to specify a wait time other than 0, issue the DIG command with the -T option. See z/OS Communications Server: IP User’s Guide and Commands for the syntax of the DIG command. Otherwise, no action is necessary.
**EZB3252I** invalid dig -T value tvalue

**Explanation**
The -T value specified on the DIG command is not valid.

**System action**
TCP/IP continues.

**Operator response**
Specify a -T value in seconds. 0 is the default.

**EZB3253I** ; invalid class specified

**Explanation**
The network class specified on the DIG command is not valid.

**System action**
TCP/IP continues.

**Operator response**
Specify a valid network class on the DIG command. DIG recognizes only the IN, CHAOS, HESIOD, and ANY network classes.
Procedure name
PrintHelp

EZB3254I ; invalid type specified

Explanation
The query type specified on the DIG command is not valid.

System action
TCPIP continues.

Operator response
Specify a valid query type, qtype on the DIG command. See z/OS Communications Server: IP User's Guide and Commands for information about the DIG command and valid query types.

System programmer response
None.

Module
DIG

Procedure name
PrintHelp

EZB3255I Missing batch file name

Explanation
The -f option was specified on the DIG command with no data set name.

System action
TCPIP continues.

Operator response
Specify a batch data set name on the -f option of the DIG command and reissue the command. z/OS Communications Server: IP User's Guide and Commands contains information about the DIG command and -f option.

System programmer response
None.

Module
DIG

Procedure name
PrintHelp

EZB3256I invalid dig -x option option

312 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**Explanation**  
The -x option specified on the DIG command is not valid.

**System action**  
TCPIP continues.

**Operator response**  
Specify a valid dotted decimal notation IP address. *z/OS Communications Server: IP User's Guide and Commands* describes the DIG command and the -x option.

**System programmer response**  
None.

**Module**  
DIG

**Procedure name**  
PrintHelp

---

**EJB3257I  no dig -p value specified**

**Explanation**  
The DIG command -p option was specified with no value. The port number given when contacting the name server must be specified with the -p option.

**System action**  
TCPIP continues.

**Operator response**  
Specify the port number given when contacting the name server on the DIG command with the -p option. The default for the Domain Name Server is 53. The -p option allows you to override this default.

**System programmer response**  
None.

**Module**  
DIG

**Procedure name**  
PrintHelp

---

**EJB3258I  invalid dig -p value $pvalue**

**Explanation**  
The -p value specified on the DIG command is not valid.

**System action**  
TCPIP continues.
**Operator response**

Specify a valid port number on the DIG -p option and reissue the command. Use the decimal port number given when contacting the name server.

**System programmer response**

None.

**Module**

DIG

**Procedure name**

PrintHelp

**EZB3259I ; invalid type⁄class specified**

**Explanation**

The query type or query class specified on the DIG command is not valid.

**System action**

TCPIP continues.

**Operator response**

Specify a valid query class and query type on the DIG command. See z/OS Communications Server: IP User's Guide and Commands for information about valid classes and types.

**System programmer response**

None.

**Module**

DIG

**Procedure name**

PrintHelp

**EZB3260I ; pflag: pfcode res: resvalue**

**Explanation**

This message displays the print flag values and the resolver options that are set for the request.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

None.
Module
DIG

Procedure name
PrintHelp

EZB3262I ; Bad server: servername - - using default server and timer opts

Explanation
The server specified is not a recognized server. DIG is using the default server and the associated timer options instead.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG

Procedure name
PrintHelp

EZB3263I ;;; FROM: hostname to SERVER: servername

Explanation
Messages are being sent from the host indicated to the server indicated.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG

Procedure name
PrintHelp

EZB3264I ;; WHEN: time
Explanation
The day, date, and time of the DIG request is displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG

Procedure name
PrintHelp

EZB3265I      ;; re_mkquery: buffer too small

Explanation
The storage buffer is too small to complete the query request.

System action
TCPIP continues.

Operator response
Define a larger buffer and reissue the DIG query request.

System programmer response
None.

Module
DIG

Procedure name
PrintHelp

EZB3271I      ;; MSG SIZE sent sendsize rcvd: recvsize

Explanation
The size of the message when sent from the host and as received by the server is displayed.

System action
TCPIP continues.
Operator response
None.

System programmer response
None.

Module
DIG

Procedure name
PrintHelp

EZB3280I ; Matching SOA found

Explanation
The authority record being searched for was located.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@LIST

Procedure name
do_zone

EZB3281I ; ListHosts: error receiving zone transfer:

Explanation
An error occurred during a zone transfer between a primary and a secondary name server.

System action
TCPIP continues.

Operator response
None.

System programmer response
Contact IBM software support services to report this error.
Module
DIG@LIST

Procedure name
do_zone

EJB32821 ; result: text, answers = value, authority = value

Explanation
This message displays with message EJB3281I and indicates the result of the zone transfer attempt, information from the answer section of the response, and the address of the authoritative name server for the response.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@LIST

Procedure name
do_zone

EJB3283X additional= number

Explanation
This message appears with message EJB3282I. Additional resources records that have not been requested, but might be useful, are displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@LIST

Procedure name
do_zone
**EZB3284I ;***Error during listing of domain**

**Explanation**
An error occurred during the listing of information about the domain name indicated in the message.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services to report this error.

**Module**
DIG@LIST

**Procedure name**
do_zone

**EZB3286I ;*** Invalid option: option**

**Explanation**
A DIG command option that was specified is not valid.

**System action**
TCPIP continues.

**Operator response**
Check the command you entered to be sure you specified the command with the correct syntax. z/OS Communications Server: IP User's Guide and Commands describes the DIG command and its syntax.

**System programmer response**
None.

**Module**
DIG@OPT

**Procedure name**
SetOption

**EZB3287I invalid timeout value: value**

**Explanation**
The time-out value specified on the DIG command is not a valid value. The time-out value specifies the number of seconds to wait before timing out of a request.
**System action**
TCPIP continues.

**Operator response**
Specify a decimal digit value for the time-out value using the DIG command.

**System programmer response**
None.

**Module**
DIG@OPT

**Procedure name**
SetOption

**EZB3288I invalid retry value: limit**

**Explanation**
The retry limit specified on the DIG command is not valid. The retry value specifies the number of times a request is sent.

**System action**
TCPIP continues.

**Operator response**
Specify a decimal digit value for the retry limit using the DIG command.

**System programmer response**
None.

**Module**
DIG@OPT

**Procedure name**
SetOption

**EZB3289I ; ***Bad char in numeric string -- ignored**

**Explanation**
DIG found a nonnumeric character in a numeric string. DIG is ignoring the character.

**System action**
TCPIP continues.

**Operator response**
None.
**System programmer response**
None.

**Module**
DIG@OPT

**Procedure name**
SetOption

**EZB3314X service_name**

**Explanation**
This message displays the service name. This message is issued for debugging purposes.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
DIG@DEBU

**Procedure name**
p_rr

**EZB3316I cms;; ->>HEADER<<<**

**EZB3317X opcode: code**

**EZB3318X , status status**

**EZB3319X , id: idname**

**Explanation**
These messages display in response to a DIG command. They indicate the operation type, for example, QUERY; the status of the request, for example, noerror, which says that the request was made correctly; and the ID of the queried Domain Name Server. If the DIG command is issued with no other parameters, this information is returned about the default server.

**System action**
TCPIP continues.

**Operator response**
None.
System programmer response
None.

Module
DIG@DEBU

Procedure name
fp_query

EZB3320I    ;; flags:
EZB3321X    qr
EZB3322X    aa
EZB3323X    tc
EZB3324X    rd
EZB3325X    ra
EZB3326X    pr
EZB3327X    res_opts: options

Explanation
Some combination of the above flags are displayed to indicate which options are set on. The flags and their meanings are as follows:

Flag    Meaning
qr      Prints the outgoing query.
aa      Accepts only authoritative responses to queries.
tc      Truncated.
rd      Recursion required.
ra      Recursion available.
pr      Uses only the primary name server for the zone.
res_opts resolver options

System action
TCP/IP continues.

Operator response
None.

System programmer response
None.
Module
DIG@DEBU

Procedure name
fp_query

EZB3328I ;; Ques: value,
EZB3329I Ans: value,
EZB3330I Auth: value,
EZB3331I Addit: value,

Explanation
This message displays the values of the question, answer, authoritative, and additional sections of the response. The question section contains the original query; the answer section contains the set of all resource records from the name server database that satisfy the query; the authoritative section contains resource records that specify the address of an authoritative name server for the query; and the additional section contains resource records that have not been explicitly requested, but could be useful.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
fp_query

EZB3332I ;; QUESTIONS:
EZB3333X ;;
EZB3334X , type = type
EZB3335X , class = class

Explanation
These messages print the question records as well as the type of query requested and the network class requested.

System action
TCPIP continues.

Operator response
None.
**Explanation**
This is the time-to-live (TTL) value for the resource record. TTL is the number of seconds that a record is valid in a cache.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

---

**Explanation**
This is the network class requested in the query and the type of query to be performed.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.
Procedure name
p_rr

EZB3338X  type

Explanation
This is the type of query to be performed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EZB3339I  ;; proto: number
EZB3340X  , port: number

Explanation
These messages indicate the network protocol and the port number for the domain name identified by the address record.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EZB3341X  string
EZB3342X  string
Explanation
These messages indicate the central processing unit type and the operating system of a node. This information is part of the host information record.

System action
TCPIP continues.

Operator response
None.

System programmer response

Module
DIG@DEBU

Procedure name
p_rr

EZB3343X  (serialvalue;serial
EZB3344X  refreshvalue;refresh
EZB3345X  retryvalue;retry
EZB3346X  expirevalue;expire
EZB3347X  minimvalue);minim

Explanation
These messages are part of the authoritative section of the response to a DIG query that requested authority records. The following list defines each of the values:

- Serial is the serial number of the zone database.
- Refresh is the refresh interval, or the length of time, in seconds, you must allow between the refreshing of a database from a remote name server.
- Retry is the retry interval that indicates the length of time, in seconds, you must allow before trying a failed refresh again.
- Expire is the expiration time-to-live that indicates the maximum time for records to be valid in the zone database.
- Minim is the minimum time-to-live that indicates the minimum time for records to be valid in the zone database.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
Explanation
This message identifies a host that can act as a mail exchange for the domain specified in the domain name field. A mail exchange runs a mail agent that delivers or forwards mail for the domain name specified in the first field of the resource record.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EZB3348X  term

Explanation
This message displays the group ID associated with the resource record.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EZB3349X  term

Explanation
This message displays the group ID associated with the resource record.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EZB3350X  string string (
Explanation
This message indicates the IP address and protocol names for the resource record.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EZB3351X string number ( 

Explanation
This message indicates the protocol name and number for the resource record.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EZB3352I number

Explanation
This message indicates the number of protocol numbers for services stored in the well-known services record.

System action
TCPIP continues.
EZB3353X  First number bytes of hex data:

Explanation
This message displays the number of bytes that are displayed following this message. This message is issued when debugging is requested.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EZB3354I  data

Explanation
This is the data in a resource record that appears in an unspecified format (binary format). Message EZB3353X tells the number of bytes of the data that are displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
Explanation
This is the default time-to-live (TTL) value. TTL is the number of seconds that a record is valid in a cache.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EJB3355X  ;; deftimetolive

Explanation
The query packet is not the standard size.

System action
TCPIP continues.

Operator response
Resubmit the query request.

System programmer response
None.

Module
DIG@DEBU

Procedure name
p_rr

EJB3356I  ;; packet size error

Explanation
This is the default time-to-live (TTL) value. TTL is the number of seconds that a record is valid in a cache.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
Explanation
This message precedes the display of records in the answers section of a query response.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
fp_query

EZB3360I ;; AUTHORITY RECORDS:

Explanation
This message precedes the display of records in the authoritative section. These resource records specify the address of an authoritative name server for the query.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG@DEBU

Procedure name
fp_query

EZB3361I ;; ADDITIONAL RECORDS:

Explanation
This message precedes the display of records in the additional section of a query response.

System action
TCPIP continues.
**Operator response**
None.

**System programmer response**
None.

**Module**
DIG@DEBU

**Procedure name**
fp_query

**EZB3362I ;<<<<DiG version >>>**

**Explanation**
This message displays the version of the Domain Information Groper (DIG) currently in use on the system.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
DIG

**Procedure name**
main

**EZB3363X defwell-known-service**

**Explanation**
The service displayed is the default well-known-service name.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.
Module
DIG@DEBU

Procedure name
p_rr

EZB3364I  dig.help

Explanation
The help command has been issued and the DIG.HELP data set is being opened.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG

Procedure name
PrintHelp

EZB3365I  file open

Explanation
The help data set is now open.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
DIG

Procedure name
main

EZB3366I  - August 30, 1990

EZB3367I  - dig @server domain query-type query-class
EZB3368I  +query-option -dig-option comment

EZB3369I  server - a domain name or a dot-notation Internet address

EZB3370I  default: nsinteraddr parm address defined in

EZB3371I  dataset TCPIP.DATA

EZB3372I  DiG Defaults: Name server address in DIG.ENV dataset

EZB3373I  domain - request domain information for domain name

EZB3374I  query-type - a, any, cname, hinfo, mx, ns, ptr, soa, wks

EZB3375I  query-class - IN (Internet), CHAOS(obsolete), HESIOD

EZB3376I  +query option

EZB3377I  <no>debug (deb) turn on/off debugging mode <deb>

EZB3378I  <no>d2 turn on/off extra debugging mode <nod2>

EZB3379I  <no>recurse (rec) use/don't use recursive lookup <rec>

EZB3380I  retry=# (ret) set number of retries to # <4>

EZB3381I  time=# (ti) set timeout length to # seconds <4>

EZB3382I  <no>ko keep open option (implies vc) <noko>

EZB3383I  <no>vc use/don't use virtual circuit <novc>

EZB3384I  <no>defname use/don't use default domain name <novc>

EZB3385I  <no>search (sea) use/don't use domain search list <sea>

EZB3386I  domain=NAME (do) set default domain name to NAME

EZB3387I  <no>ignore (i) ignore/don't ignore trunc. errors <noi>

EZB3388I  <no>primary (pr) use/don't use primary server <nopr>

EZB3389I  <no>aaonly (aa) authoritative query only flag <noaa>

EZB3390I  <no>sort (sor) sort resource records <nosor>

EZB3391I  <no>cmd echo parsed arguments <cmd>

EZB3392I  <no>stats (st) print query statistics (RTT,etc) <st>

EZB3393I  <no>Header (H) print basic header <H>

EZB3394I  <no>header (he) print header flags <he>

EZB3395I  <no>ttlid (tt) print TTLs <tt>

EZB3396I  <no>cl print class info <nocl>

EZB3397I  <no>qr print outgoing query <noqr>

EZB3398I  <no>reply (rep) print reply <rep>

EZB3399I  <no>ques (qu) print question section <qu>

EZB3400I  <no>answer (an) print answer section <an>

EZB3401I  <no>author (au) print authoritative section <au>

EZB3402I  <no>addit (ad) print additional section <ad>

EZB3403I  pfdef set to default print flags

EZB3404I  pfmin set to minimal default print flags

EZB3405I  pfset=# set print flags to #
EZB3406I  (# can be hex/octal/decimal)
EZB3407I  pfand=# bitwise and print flags with #
EZB3408I  pfor=# bitwise or print flags with #
EZB3409I  - dig option
EZB3410I  -x dot-notation-address
EZB3411I  -f file for dig batch mode
EZB3412I  -T Time in seconds between start of successive queries
EZB3413I  -p Port number
EZB3414I  -P ping-string
EZB3415I  -t query-type
EZB3416I  -c query-class
EZB3417I  -envsav This flag specifies that the dig environment
EZB3418I  (defaults, print options, etc.), after all of
EZB3419I  the arguments are parsed, should be saved to a
EZB3420I  file to become the default environment. DiG.env
EZB3421I  is created in the current working directory.
EZB3422I  -envset This flag only affects batch query runs. When
EZB3423I  -envset is specified on a line in a dig batch file
EZB3424I  the dig environment after the arguments are parsed,
EZB3425I  becomes the default environment for the duration of
EZB3426I  the batch file, or until the next line which
EZB3427I  specifies -envset.
EZB3428I  --no>stick This flag only affects batch query runs. It
EZB3429I  specifies that the dig environment (as read
EZB3430I  initially or set by -envset switch) is to be
EZB3431I  restored before each query (line) in a dig
EZB3432I  batch file.
EZB3433I  %comment - included argument that is not parsed

Explanation

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
**EZB3434I**  Can not read DIG environmental data set. Defaults used.

**Explanation**
This message is displayed if a DIG command was invoked with the -envsav option from an unsupported release of Communications Server. This option is not compatible with the current supported releases of Communications Server.

**System action**
No query options saved from the DIG command prior to Communications Server for V1R2 are restored to a Communications Server for V1R2 or later release of the DIG command.

**Operator response**
To create an environmental data set in a Communications Server for V1R2 or later format, use the -envsav option with any DIG command that has the query options required. See z/OS Communications Server: IP User's Guide and Commands for information about the TSO DIG command.

**System programmer response**
none

---

**EZB3483E**  Couldn't open output file

**Explanation**
NSDBLOAD encountered an error opening its output file. The program ends with exit code 99.

**System action**
The name server exits.

**Operator response**
Verify that the name server has read/write access to the file.

**System programmer response**
None.
Procedure name
main

**EZB3484E** Error(s) occurred reading master datafile.

**Explanation**
NSDBLOADz encountered an error while processing the master data file.

**System action**
The name server ends.

**Operator response**
Check the data syntax of the master data file.

**System programmer response**
Contact IBM software support services.

**Module**
NSDBLOAD

**Procedure name**
main

**EZB3487I** Error occurred deleting table info.

**Explanation**
NSDBLOADz encountered an error deleting SQL table information.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NSDBLOAD

**Procedure name**
main

**EZB3489I** sql statement sqlcode sqlcode

**Explanation**
NSDBLOAD was unable to process the indicated SQL command.
System action
NSDBLOAD continues.

Operator response
Use the sqlcode that is displayed in this message and http://www.ibm.com/support/knowledgecenter/SSEPH2/welcome to determine the cause of the error and respond as indicated.

System programmer response
None.

Module
NSDBLOAD

Procedure name

EZB3490I  time: Delete all data from name table

Explanation
NSDBLOADz is deleting all data from the indicated table.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
NSDBLOAD

Procedure name
deleteinfo

EZB3491I  time: Deleted all data in sql table name

Explanation
NSDBLOADz has deleted all data in the indicated SQL table.

System action
TCPIP continues.

Operator response
None.
System programmer response
None.

Module
NSDBLOAD

Procedure name
deleteinfo

EZB3493E Couldn’t read output resource record file

Explanation
NSDBLOADz encountered an error opening its output file. The program ends with exit code 99.

System action
The NSDBLOAD program ends.

Operator response
Check for the name server’s access to the file.

System programmer response
None.

Module
NSDBLOAD

Procedure name
insertsql

EZB3495W error invalid class class

Explanation
The NSDBLOAD program encountered an incorrect class specification in an input record. The record is ignored.

System action
TCP/IP continues.

Operator response
Check for the correct CLASS of the defined resource record.

System programmer response
None.

Module
NSDBLOAD
Procedure name
insertsql

EZB3496W   error invalid type type

Explanation
The NSDBLOAD program encountered an incorrect type specification in an input record.

System action
The record is ignored.

Operator response
Check for the correct TYPE of the defined resource record.

System programmer response
None.

Module
NSDBLOAD

Procedure name
insertsql

EZB3497I   time: Finished adding data to table ....

Explanation
The NSDBLOAD program has finished adding data to its SQL table.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
NSDBLOAD

Procedure name
insertsql

EZB3498I   time: Update statistics on the name table

Explanation
The NSDBLOAD program updates the statistics on this table.
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NSDBLOAD

**Procedure name**
insertsql

**EZB3499I**  
*time: Data base update completed.*

**Explanation**
The NSDBLOAD program has completed the database update.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NSDBLOAD

**Procedure name**
readmaster

**EZB3501E**  
*Couldn't read input master file*

**Explanation**
The NSDBLOAD program encountered an error opening the resource record file previously generated. The program ends with exit code 99.

**System action**
The name server ends.

**Operator response**
Check the read/write access on the input file for the current user ID.
System programmer response
Check for the existence of a resource record file.

Module
NSDBLOAD

Procedure name
insertsql

EZB3503E Origin definition error

Explanation
The NSDBLOAD program encountered a $ORIGIN statement in the input file without an accompanying definition.

System action
TCPIP continues.

Operator response
Correct the $ORIGIN input statement to NSDBLOAD.

System programmer response
None.

Module
NSDBLOAD

Procedure name
readmaster

EZB3504I time: New origin origin

Explanation
The NSDBLOAD program encountered a $ORIGIN statement with the specified origin in its input file.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
NSDBLOAD
Procedure name
readmaster

EZB3505I  time: @ - current origin origin

Explanation
The NSDBLOAD program is using the specified origin.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
NSDBLOAD

Procedure name
readmaster

EZB3506E  Origin has not been defined

Explanation
A relative domain name defined in a master file requires a domain origin suffix.

System action
TCPIP continues.

Operator response
Correct the master file to include a $ORIGIN domain statement.

System programmer response
None.

Module
NSDBLOAD

Procedure name
readmaster

EZB3509W  Format error 1: line

Explanation
The NSDBLOAD program encountered a syntax error in the input file. The line causing the error is displayed and ignored.
**System action**
TCPIP continues.

**Operator response**
Check the TYPE field in the defined resource record.

**System programmer response**
None.

**Module**
NSDBLOAD

**Procedure name**
readMaster

**EZB3510W Format error 2: line**

**Explanation**
The NSDBLOAD program encountered a syntax error in the input file. The line causing the error is displayed and ignored.

**System action**
TCPIP continues.

**Operator response**
The required TYPE field is missing from the resource record.

**System programmer response**
None.

**Module**
NSDBLOAD

**Procedure name**
readMaster

**EZB3511W Format error 3: line**

**Explanation**
The NSDBLOAD program encountered a syntax error in the input file. The line causing the error is displayed and ignored.

**System action**
TCPIP continues.

**Operator response**
The required TYPE field is missing from the resource record.
System programmer response
None.

Module
NSDBLOAD

Procedure name
readMaster

EZB3512W Obsolete root definition replace with ‘.’

Explanation
The NSDBLOAD program encountered an obsolete root definition (..) in the input file. The NSDBLOAD program ignores this error.

System action
TCPIP continues.

Operator response
Change the root definition from ‘..’ to ‘.’.

System programmer response
None.

Module
NSDBLOAD

Procedure name
readmaster

EZB3513W Invalid wild card definition

Explanation
The NSDBLOAD program encountered an incorrect wildcard definition in the input file.

System action
TCPIP continues.

Operator response
Correct the input file. See RFC 1035 for the correct wild card entry. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

System programmer response
None.

Module
NSDBLOAD
Procedure name
readmaster

EZB3514W   Mailbox in SOA data field is specified incorrectly.

Explanation
The NSDBLOAD program encountered an incorrectly specified RNAME field in an SOA RR RDATA field while reading the master data file.

System action
TCPIP continues.

Operator response
See RFC 1034 for the correct specification of the RNAME field. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

System programmer response
None.

Module
NSDBLOAD

Procedure name
readmaster

EZB3515I   Drop index sql table

Explanation
NSDBLOAD will preform an SQL drop index command against the table displayed in the message.

System action
NSDBLOAD performs the SQL drop index command and continues.

Operator response
None.

System programmer response
None.

Module
NSDBLOAD

Procedure name
dropindex

EZB3516I   SQLcommand

Explanation
The name server will perform a create index SQL command against the SQL table displayed in the message.
System action
The NSDBLOAD performs the create index command and continues.

Operator response
None.

System programmer response
None.

Module
NSDBLOAD

Procedure name
creatindex

EZB3517I  *** WARNING: OWNER of SQL table is not NAMESEV. **
EZB3518I  *** Ensure view exists on owner.nstable. **

Explanation
The current name server SQL table is not owned by NAMESRV.

System action
NSDBLOAD continues.

Operator response
Make sure that the owner specified in the message has view authority for the table.

System programmer response
None.

Module
NSDBLOAD

Procedure name
whichtable

EZB3519I  Select tcurrent

Explanation
NSDBLOAD is selecting a table to receive SQL data.

System action
NSDBLOAD continues.

Operator response
None.
System programmer response
None.

Module
NSDBLOAD

Procedure name
whichtable

**EZB3520I**  
**PREPARE SQL error: sqlcode**

Explanation
NSDBLOAD did a PREPARE of the SQL Db2® table, and SQL returned the code that is displayed in the message.

System action
NSDBLOAD continues.

Operator response
Use the *sqlcode* that is displayed in this message and http://www.ibm.com/support/knowledgecenter/SSEPH2/welcome to determine the cause of the error and respond as indicated.

System programmer response
None.

Module
NSDBLOAD

Procedure name
whichtable

**EZB3521I**  
**Declare SQL error: sqlcode**

Explanation
NSDBLOAD did a DECLARE of the SQL Db2 table and received the return code that is displayed in the message.

System action
NSDBLOAD continues.

Operator response
Use the *sqlcode* that is displayed in this message and http://www.ibm.com/support/knowledgecenter/SSEPH2/welcome to determine the cause of the error and respond as indicated.

System programmer response
None.

Module
NSDBLOAD
**Procedure name**

whichtable

**EZB3522I**  Open SQL error: sqlcode

**Explanation**

NSDBLOAD did an OPEN of the SQL Db2 table and received the return code that is displayed in the message.

**System action**

NSDBLOAD continues.

**Operator response**

Use the sqlcode that is displayed in this message and SGL/DS Messages and Codes to determine the cause of the error and respond as indicated.

**System programmer response**

None.

**Module**

NSDBLOAD

**Procedure name**

whichtable

**EZB3523I**  Table ttable not defined in nstable.

**EZB3524I**  using default table sqltable

**Explanation**

NSDBLOAD performed an SQL FETCH command. The table displayed in the message was not defined in the nstable.

**System action**

NSDBLOAD uses the default table and continues.

**Operator response**

Verify that the table specified in the NSDBLOAD command was valid and it has been defined in the DNSTABLE data set.

**System programmer response**

None.

**Module**

NSDBLOAD

**Procedure name**

whichtable

**EZB3525I**  Using sql table table
The table displayed is being used by NSDBLOAD.

NSDBLOAD continues.

None.

None.

NSDBLOAD

whichtable

EZB3534I NSDBLOAD reads a master dataset, as defined in RFC1034,
EZB3535I generates resource records, and inserts them into an SQL
EZB3536I table. Multiple name servers can execute simultaneously
EZB3537I on the same system assuming they are using separate ports
EZB3538I or are executing on different TCPIP service machines.
EZB3539I When inserting data into an SQL table owned by NAMESRV the
EZB3540I above SQL names are permitted. When inserting into an SQL
EZB3541I table NOT owned by namesrv, the SQL tablename must be fully
EZB3542I qualified. If a view is defined on nstable, NSDBLOAD will
EZB3543I determine from the status field which table is current (0|1)
EZB3544I Format: nsdbload [db2-subsystem sqltable input-dataset
EZB3545I output-dataset]
EZB3546I where: db2name – SQL subsystem name
EZB3547I sqltable – the table owner defaults to NAMESRV.
EZB3548I Any of the following forms are permitted:
EZB3549I cache, namesrv.cache, or namesrv.cache0.
EZB3550I input-dataset – master data dataset
EZB3551I output-dataset – resource record dataset

NSDBLOAD displays this help text if you specified a question mark as the only command line parameter.

TCPIP continues.
Operator response
None.

System programmer response
None.

Module
NSDBLOAD

Procedure name
makelower

EZB3560I time: Do you want to create resource records from a master file (y/n)?

Explanation
The NSDBLOAD command is prompting you to determine whether the resource records should be created from a master file or the program should end now.

System action
TCPIP continues.

Operator response
Enter Y or N.

System programmer response
None.

Module
NSDBLOAD

Procedure name
main

EZB3561I time: Do you want to delete all resource records from db(y/n)?

Explanation
The NSDBLOAD command is prompting you to determine whether all resource records should be deleted from the database.

System action
TCPIP continues.

Operator response
Enter Y or N.

System programmer response
None.
Module
NSDBLOAD

Procedure name
main

EJB3562I  time: Do you want to insert the RR in the sqltable now(y/n)?

Explanation
The NSDBLOAD command is prompting you to determine whether the resource records defined in the input parameters should be inserted into the SQL tables now.

System action
TCPIP continues.

Operator response
Enter Y or N.

System programmer response
None.

Module
NSDBLOAD

Procedure name
main

EJB3563I  time: Do you want translate all data to lower case?(y/n)

Explanation
The NSDBLOAD command is prompting you to determine whether all input data should be converted to lowercase.

System action
TCPIP continues.

Operator response
Enter Y or N.

System programmer response
None.

Module
NSDBLOAD

Procedure name
insertsql

EJB3564I  time: Do you want translate all data to upper case?(y/n)
Explanation
NSDBLOAD is prompting you to determine whether all input data should be translated to uppercase.

System action
TCPIP continues.

Operator response
Enter Y or N.

System programmer response
None.

Module
NSDBLOAD

Procedure name
insertsql

EZB3565I time: Drop index table

Explanation
NSDBLOAD is issuing an SQL DROP INDEX command for this table.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
NSDBLOAD

Procedure name
dropindex

EZB3566I RDATA Error: resource record

Explanation
The resource record displayed is missing the RDATA field.

System action
NLDBLOAD increments an error counter and continues.
Operator response
Correct the resource record of the name server SQL table.

System programmer response
None.

Module
NSDBLOAD

Procedure name

EJB3825T  name/udp: unknown service

Explanation
A UDP port number for the service name was not assigned in the tcpip.v3r1.ETC.SERVICES data set.

System action
NCPROUTE exits.

Operator response
None.

System programmer response
Verify that the tcpip.v3r1.ETC.SERVICES data set has two entries in the form:

ncproute  
  port/udp
router  
  port/udp

The entries must start in column 1 and be in lowercase. Verify that the NCPROUTE service port number is the port being used by the NCP clients. The port value defined in the UDPPORT= keyword on the IPOWNER statement in the NCP generation definition must match the NCPROUTE service port. The default is UDP port 580.

The reserved router service port number is 520 and is required for the NCPROUTE transport of RIP packets to NCP clients, which are responsible for broadcasting the packets to other RIP routers. The router service port number cannot be overridden as a result of an NCP restriction.

Also, verify that port 580 has been reserved for NCPROUTE under the PORT statement in the tcpip.v3r1.PROFILE.TCPIP data set.

Module
NRMAIN

Procedure name
main

EJB3826I  Port port assigned to name

Explanation
NCPROUTE will listen for traffic from NCP clients on the specified port port assigned to service name.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
If communications cannot be established with a client, compare NCPROUTE’s port number with the value specified in the client’s NCP generation definition. The port value defined in the UDPPORT= keyword on the IPOWER statement in the NCP generation definition must match.

Module
NRMAIN

Procedure name
main

EZB3827T   Terminating since clients require the socket

Explanation
NCPROUTE attempted to open a socket on a well known or user-defined port but the open was not successful, or the socket could not be bound to an IP address and port number. Clients will not be able to communicate with NCPROUTE because a socket is not available.

System action
NCPROUTE ends abnormally.

Operator response
None.

System programmer response
Examine previous messages to determine the nature of the error as indicated by a detailed tcperror() library message. Correct the problem as indicated by error. See the z/OS XL C/C++ Runtime Library Reference for more information about socket() function errors.

Module
NRMAIN

Procedure name
Main

EZB3828T   Usage: NCPROUTE parameters

Explanation
Incorrect parameters were passed to NCPROUTE.

System action
NCPROUTE ends abnormally.
Operator response
None.

System programmer response
Verify that the parameters are correct. The parameters are case-sensitive and must be separated by spaces. See z/OS Communications Server: IP Configuration Reference for more information.

Module
NRMAIN

Procedure name
main

EZB3829I   Waiting for incoming packets

Explanation
NCPROUTE is waiting for datagrams from NCP clients. Each time at which NCPROUTE finishes processing an event, such as an incoming datagram or a timer that expires, NCPROUTE issues this message and waits for the next event. These messages should occur at least once every 30 seconds, but will increase in frequency as the server performs more work.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Check to see if the assigned port number for NCPROUTE matches the NCP clients' generation definitions.
If six or more of these messages occur consecutively, NCPROUTE is not receiving any datagrams from NCP clients. Verify that a client has an established session with NCPROUTE, and if one exists, examine the status of the client's interfaces.

Module
NRMAIN

Procedure name
main

EZB3830E   The main select was interrupted:

Explanation
An error occurred while NCPROUTE was waiting for an event to occur. A more detailed tcperror() library message follows.

System action
NCPROUTE continues, unless an IUCV error occurred.
Operator response

None.

System programmer response

Verify that TCPIP is running.

Module

NRMAIN

Procedure name

Main

EJB3831I  Send delayed dynamic update

Explanation

A routing update, which had been delayed to prevent packet storms, has been transmitted. This occurs 2–5 seconds after a dynamic update has been issued.

System action

NCPROUTE continues.

Operator response

None.

System programmer response

None.

Module

NRMAIN

Procedure name

Main

EJB3832E  While receiving a packet from a client:

Explanation

An error occurred while attempting to receive a packet from a client. A more detailed tcperror() library message follows.

System action

NCPROUTE continues, and the incoming packet is discarded. If an IUCV error occurs, NCPROUTE will end.

Operator response

None.

System programmer response

See the next generated error message and correct the error.
Module
NRMAIN

Procedure name
process

EZB3833E While receiving a packet from the SNMP agent: agent

Explanation
An error occurred while attempting to read a packet from the SNMP agent. A more detailed tcperror() library message follows. If the connection with the Agent has been reset, the SNMP agent will be terminated and incoming SNMP requests will be ignored.

System action
NCPROUTE continues. The SNMP packet is discarded. If an IUCV error occurs, NCPROUTE will end.

Operator response
None.

System programmer response
If the connection with the SNMP agent has been reset, restart the SNMP daemon (OSNMPD).

Module
NRMAIN

Procedure name
read_dpi

EZB3834I ****************************************

Explanation
Two of these banners enclose a message that may need attention when viewing the output. The severity of the enclosed message is indicated.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Read the enclosed message and, if necessary, resolve the indicated situation.

Module
various.

Procedure name
various.
**EZB3835T** Invalid parameter: *parameter*

**Explanation**
An incorrect parameter was passed from the tcpip.v3r1.SEZAINST(NCPROUT) start proc JCL. The parameter could be passed from the command line parameters or from the default parameter list in the start proc JCL.

**System action**
NCPROUTE exits.

**Operator response**
None.

**System programmer response**
Correct the parameter from the command line parameters or in the default parameter list of the start proc JCL.

**Module**
NRMAIN

**Procedure name**
main

**EZB3836W** The SNMP agent has terminated

**Explanation**
The socket used to communicate with the SNMP agent has been reset. This implies that the SNMP agent has ended. Without this socket, SNMP requests are not processed.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
If the SNMP agent (OSNMPD) is not running, restart the agent. NCPROUTE will attempt to reestablish communications with the agent after each new SNMP request is received, so no further action is required. If the agent is running, contact IBM software support services.

**Module**
NRMAIN

**Procedure name**
read_dpi

**EZB3837W** SNMP requests will be ignored until it is restarted.

**Explanation**
Because communication with the SNMP agent is not possible, incoming SNMP requests are ignored until a connection has been established.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
Start the SNMP agent (OSNMPD). NCPROUTE will attempt to establish communication with the agent for each SNMP packet received until a connection is established.

Module
NRMAIN

Procedure name
read_dpi

EZB3838W  An SNMP DPI packet arrived from a non-internet machine

Explanation
An SNMP DPI packet was received from a non-Internet network. SNMP DPI traffic is only accepted from machines running TCPIP. The incoming packet is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Locate the machine generating the SNMP DPI packet and have it stopped. Ignore these messages.

Module
NRMAIN

Procedure name
read_dpi

EZB3839T  A socket could not be created:

Explanation
NCPROUTE could not open a new socket. With the socket unavailable, the NCP clients will not be able to establish communications with NCPROUTE. A more detailed tcperror() library message follows.

System action
NCPROUTE ends abnormally.

Operator response
None.
System programmer response
Verify that TCPIP is active, that another program is not using NCPROUTE's port, that the well-known port has been reserved in the PORT statement of hlq.PROFILE.TCPIP data set, and that the user-defined port has been specified in hlq.ETC.SERVICES data set. Correct the problem as indicated by the error in the detailed tcperror() library message. See the z/OS XL C/C++ Runtime Library Reference for more information about socket() function errors.

Module
NRMAIN

Procedure name
getsocket

EZB3840T Broadcasting cannot be enabled on the socket:

Explanation
NCPROUTE cannot enable the socket for broadcasting. NCPROUTE must be able to broadcast over interfaces which support broadcasting in order to communicate with the NCP clients. A more detailed tcperror() library message follows.

System action
NCPROUTE ends abnormally.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRMAIN

Procedure name
getsocket

EZB3841T The socket bind failed

Explanation
NCPROUTE was unable to associate an IP address and port number to the newly created socket. Another application could be using the port. A more detailed tcperror() library message follows.

System action
NCPROUTE ends abnormally.

Operator response
None.
**System programmer response**

Use the NETSTAT ALLCONN command to verify that there is no other application using the NCPROUTE port number. Look for a port in the “Local Socket” column which matches NCPROUTE’s port in the hlq.ETC.SERVICES data set. You should reserve this port for NCPROUTE’s exclusive use by adding an entry to the PORT statement in the hlq.PROFILE.TCPIP data set.

**Module**

NRMAIN

**Procedure name**

Getsocket

**EJB3842W Hello from existing client client**

**Explanation**

An NCP client with a current session has entered a reset state and has started to send Hello packets in an attempt to establish a session with NCPROUTE. This client might have been shut down and restarted. The current session ends and a new one is started.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

NRPDUS

**Procedure name**

Pdu_in

**EJB3843W Status from a session-less client client**

**Explanation**

A client has transmitted an interface status change PDU without first establishing a session with NCPROUTE. This is a protocol violation. The PDU is discarded. Also, the client is added to a “Client Protocol Violation List” and no further error messages will be issued for this client until it successfully establishes a session. At that point, NCPROUTE will accept the client, and the client’s name will be removed from the list.

**System action**

NCPROUTE continues.

**Operator response**

None.
System programmer response
Restart the client NCP. If the problem persists, contact IBM software support services.

Module
NRPDUS

Procedure name
pdu_in

EZB3844W This datagram is being ignored

Explanation
A datagram was received and some outstanding error prevents NCPROUTE from operating on the packet. The most likely cause is that a protocol error occurred between the client and NCPROUTE. The datagram is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Identify the cause of the error from a previous error message and correct the problem. The client must be reset so that a new session is established.

Module
NRPDUS

Procedure name
pdu_in

EZB3845W Transport from session-less client client

Explanation
A client NCP has transmitted a Transport PDU without first establishing a session with NCPROUTE. This is a protocol violation. The PDU is discarded. Also, the client is added to a “Client Protocol Violation List” and no further error messages will be issued for this client until it successfully establishes a session. At that point, NCPROUTE will accept the client and the client's name will be removed from the list.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Restart the client NCP. If the problem persists, contact IBM software support services.
**Module**
NRPDUS

**Procedure name**
pdu_in

**EZB3846W**  InactList from session-less client *client*

**Explanation**
A client NCP has transmitted an inactive interface PDU without first establishing a session with NCPROUTE. This is a protocol violation. The PDU is discarded. Also, the client is added to a “Client Protocol Violation List” and no further error messages will be issued for this client until it successfully establishes a session. At that point, NCPROUTE will accept the client and the client's name will be removed from the list.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Restart the client NCP. If the problem persists, contact IBM software support services.

---

**Module**
NRPDUS

**Procedure name**
pdu_in

**EZB3847W**  Protocol Violation: client *client* sent

**Explanation**
A foreign machine sent a packet to the NCPROUTE port that does not contain a valid type field in the packet header. This is probably not an NCP, but another machine on the network. The PDU is discarded. Also, the client is added to a “Client Protocol Violation List” and no further error messages will be issued for this client until it successfully establishes a session. At that point, NCPROUTE will accept the client and the client's name will be removed from the list.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Identify the source of the packet from the displayed IP address and correct the problem. If the machine in question is an NCP, verify that it is loaded with the correct software and reset it. If the machine is an NCP and resetting it does not correct the problem, contact IBM software support services.
**EZB3848W** An excessive number of clients (#) have been issued protocol violations. Further warning messages will be suppressed.

**Explanation**
A large number of clients have committed protocol violations and have not subsequently established sessions with NCPROUTE. No further “Warning” messages are issued. NCPROUTE assumes that the problems have been noticed by this point and that steps are being taken to correct them.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Verify that the most recent versions of NCPROUTE and the NCP drivers are installed. Verify that previous protocol errors on individual client NCPs were investigated and corrective actions were taken. Contact IBM software support services if the protocol violations continue.

---

**EZB3850E** Status change for unknown interface *interface*

**Explanation**
NCPROUTE received a status change request PDU from the NCP client for an interface that was unknown. Either NCPROUTE did not completely read in the NCP client’s Routing Information Table (RIT), or the RIT was not built correctly during NCP generation. Another possible cause is that the NCP client dynamically added the interface to its tables. Dynamically-added interfaces are not currently supported by NCPROUTE. The Status PDU is discarded.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Check the interface configuration messages created in the console log at install time to verify that the interfaces defined in the NCP client’s generation match the interfaces processed by NCPROUTE. If a mismatch is found, verify that the RIT is built correctly. If the problem cannot be corrected, contact IBM software support services.

---

EZW3xxx messages 365
Module
NRPDUS

Procedure name
recv_status

EZB3851I Accepting bad client client

Explanation
A client that committed a protocol violation earlier has successfully established a session and any new protocol violations are reported.

System action
The client is removed the “Client Protocol Violation List”. With the client removed from this list, NCPROUTE resumes reporting any new protocol violation errors.

Operator response
None.

System programmer response
None.

Module
NRPDUS

Procedure name
forgive

EZB3855I NCP Add out to client Route to address1 via interface interface to address2 Metric: metric, Type type, Subnetmask mask

Explanation
An "Add" PDU is being sent to a client, which causes a route to be added to its IP route tables.

address1 is the destination IP address of the route.
The interface is the name given to the interface during NCP generation.
address2 is the IP address of an intermediate router or zero if the destination is directly connected.
The metric is the relative cost of using this route as opposed to another route.
The type is either Host, Subnet, or Network and indicates the route type being added.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
Module
NRPDUS

Procedure name
nr_do_add

EZB3856I Pending delete for interface interface

Explanation
NCPROUTE received a status change request from the NCP client to delete the specified interface from its interface tables. The NCP client had deleted the interface from its configuration as a result of dynamic reconfiguration. NCPROUTE puts the interface in pending delete state so that routing outages can be reported to other interfaces and the incorrect addition of routes in routing responses can be prevented. NCPROUTE will continue to remove the specified interface from its tables until either the routes are timed out or a status change request to add a new interface with the same IP address is received.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRPDUS

Procedure name
recv_status

EZB3857I Adding new interface interface

Explanation
NCPROUTE received a status change request from the NCP client to add a new interface to its interface tables. The NCP client had added the interface to its configuration as a result of dynamic reconfiguration. NCPROUTE will manage routes for the new interface.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
**Procedure name**
recv_status

**EZB3858E  Unknown status change request**

**Explanation**
NCPROUTE received an invalid status change request from the NCP client. The Status PDU is discarded.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

**Module**
NRPDUS

**Procedure name**
recv_status

**EZB3859I  Deleting interface interface**

**Explanation**
NCPROUTE is deleting the specified interface from its interface tables. The interface was previously in pending delete state so that routing outages could be reported to other interfaces and the incorrect addition of routes in routing responses could be prevented. The deletion occurs when either the routes attached to the interface have timed out or a status change request to add a new interface with the same IP address is received.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRPDUS

**Procedure name**
recv_status

**EZB3860E  The transmission of an “Add” to client client failed**
Explanation
TCPIP detected that the "Add" PDU could not be delivered successfully. The add request, for adding a route to the NCP client's routing table, is not performed. A more detailed tcperror() library message follows.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Examine the error message that follows and proceed according to the recommendations.

Module
NRPDUS

Procedure name
nr_do_add

EZB3862I NCP_Add out to client Route to address1 via interface interface to address2 Metric: metric,
Type type, Subnetmask mask

Explanation
A "Delete" PDU is being sent to the named client. The address is the destination of the route to be deleted, and the type is either Host, Subnet or Network and is used by the client to locate the correct route table from which to delete the route.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRPDUS

Procedure name
nr_do_delete

EZB3863S RIP2 authentication key exceeds maximum allowed or contains unsupported characters

Explanation
The RIP Version 2 authentication key, specified on the RIP2_AUTHENTICATION_KEY entry in the NCPROUTE profile data set, or on the options statement entry in the NCP client's gateways data set for an interface, is invalid. The authentication key may have contained unsupported characters or have exceeded the maximum 16 characters allowed. The authentication key is ignored and no authentication check is performed.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the NCPROUTE profile or the NCP client's GATEWAYS data set.

Module
NRMAIN, NRSTART

Procedure name
read_profile, ParseOptions

EJB3864E The transmission of a ‘Delete’ to client client failed

Explanation
TCPIP detected that the "Delete" PDU could not be delivered to an NCP client. The delete request, for deleting a route from the NCP client's routing table, is not performed. A more detailed tcperror() library message follows.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Examine the error message that follows and proceed according to the recommendations.

Module
NRPDUS

Procedure name
nr_do_delete

EJB3865T Out of memory during transport

Explanation
NCPROUTE has used all available storage while attempting to send a "Transport" PDU to a client. In this condition, NCPROUTE can no longer communicate with its clients.

System action
NCPROUTE exits.

Operator response
None.
System programmer response
Increase NCPROUTE's region size and restart. Take into consideration that storage requirements are based on the number of clients being served and the number of routes being managed. If the problem still cannot be resolved, contact IBM software support services.

Module
NRPDUS

Procedure name
nr_do_transport

EZB3866E  The transmission of a ‘Transport’ to client client failed

Explanation
TCPIP detected that it would be unable to deliver the “Transport” PDU to a client. The “Transport” PDU is discarded. A more detailed tcperror() library message follows.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Examine the error message that follows and proceed according to the recommendations.

Module
NRPDUS

Procedure name
nr_do_transport

EZB3867I  Acknowledge to client: Hello Received

Explanation
A client has attempted to establish a session with the server and is being updated on the status of the request. The "Hello" PDU was received and is sent immediately to quiet the client.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
Module
NRPDUS

Procedure name
recv_hello

**EZB3868I**  Acknowledge to client: RIT Loaded OK

**Explanation**
A client has attempted to establish a session with the server, and is being updated on the status of the request. The Routing Information Table (RIT) was loaded and appears usable.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

---

Module
NRPDUS

Procedure name
recv_hello

**EZB3869E**  Acknowledge to client: RIT Load Failed

**Explanation**
A client has attempted to establish a session with the server and is being updated on the status of the request. An abend occurred during the load of the Routing Information Table (RIT). The session with the server is not established. A previous message explains the cause of the abend.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Examine the abnormal end code in a previous message and correct the error. If the problem cannot be corrected, contact IBM software support services.
Procedure name
recv_hello

**EZB3870E**  Acknowledge to client: RIT ID Bad

**Explanation**
A client has attempted to establish a session with the server and is being updated on the status of the request. An error was detected in the correlation string of the Routing Information Table (RIT). The string does not match the one in the received "Hello" PDU. Most likely the NCP client has been reconfigured after a new NCP load and, as a result, the RIT ID was updated. The session with the server is not established.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Verify that the correct RIT was installed and that another data set was not inadvertently loaded earlier. The RIT should be referenced in the operating system's link list or in the DD:STEPLIB statement of the NCPROUTE catalogued procedure. If the problem cannot be corrected, contact IBM software support services.

**Module**
NRPDUS

**Procedure name**
recv_hello

**EZB3871E**  Acknowledge to client: RIT Bad

**Explanation**
A client has attempted to establish a session with the server and is being updated on the status of the request. An error was detected in the Routing Information Table (RIT) data. The session with the server is not established.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Verify that the correct RIT was installed and that another data set was not inadvertently loaded earlier. The RIT should be referenced in the operating system's link list or in the DD:STEPLIB statement of the NCPROUTE catalogued procedure. If the problem cannot be corrected, contact IBM software support services.

**Module**
NRPDUS
**Procedure name**
recv_hello

**EZB3872E  Acknowledge to client: RIT Not Found**

**Explanation**
A client has attempted to establish a session with the server and is being updated on the status of the request. The Routing Information Table (RIT) could not be found matching the NCP name in the received "Hello" PDU. The session with the server is not established.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Verify that the RIT built during NCP generation has been installed into a partitioned data set and referenced by the DD:STEPLIB statement of NCPROUTE catalogued procedure. If the problem cannot be corrected, contact IBM software support services.

**Module**
NRPDUS

**Procedure name**
recv_hello

**EZB3873E  Acknowledge to client: Unsupported Ack Type**

**Explanation**
A client has attempted to establish a session with the server, and is being updated on the status of the request. An unknown Acknowledgement type was received. The session with the server is not established.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

**Module**
NRPDUS

**Procedure name**
recv_hello

**EZB3875W  Variable subnetting not supported by client client**

374  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**Explanation**

Variable subnetting is not supported by the client and the RIP supply/receive control setting may have been set in the NCPROUTE configuration such that RIP Version 2 packets are to be sent or received over interface(s). NCPROUTE will override the control settings to RIP1 for compatibility with the NCP client configuration.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Verify the NCP software level for variable subnetting support in the NCP client. If RIP Version 2 packets are not to be used, correct the NCPROUTE profile data set or the NCP client's gateways data set to use RIP Version 1 packets.

**Module**

various

**Procedure name**

various

---

**EZB3876I Hello from new client client**

**Explanation**

An NCP client is attempting to establish a session with NCPROUTE. This is done by sending a "Hello" PDU containing the NCP name and correlation string used to verify that the correct Routing Information Table (RIT) is loaded.

**Note:** The NCP client will issue many "Hello" PDUs up to the maximum specified in the NCP gen until it successfully establishes a session with the server. Once the maximum is reached, the NCP client will repeat the cycle again after a 9-minute delay timer has expired. The delay timer is used to prevent NCP alert flooding. The NCP client's issuance of "Hello" PDUs can happen even after the server has established a session with the NCP client. For example, the NCP client could have been restarted. An attempt will be made to establish a new session.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

NRPDUS

**Procedure name**

recv_hello
EZB3877I  RIT dataset name: *ritdsname*

**Explanation**
The specified Routing Information Table (RIT) data set name, as supplied in the Hello PDU, is loaded and processed.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRPDUS

**Procedure name**
recv_hello

EZB3878I  RIT ID: *table_id*

**Explanation**
The specified ID string is used as a correlation string to verify that the correct Routing Information Table (RIT) has been loaded.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None., however, if this ID appears unusual, determine if the correlation string was overridden during NCP load.

**Module**
NRPDUS

**Procedure name**
recv_hello

EZB3879E  *timestamp*

**Explanation**
An error occurred and a full time stamp is written showing the date and time when the error occurred. While each message is timestamped, date information is omitted, and this message is issued to help those running for long periods of time.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRPDUS

Procedure name
Recv_hello

EZB3880E  Error opening RIT client1 for client client2

Explanation
The Routing Information Table (RIT), which was created during NCP generation, could not be opened. client1 is
the name of the member on which the load was attempted, client2 is the IP address of the client that is
attempting to establish a session with NCPROUTE. NCPROUTE does not establish the session.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Examine the abend code and reason code in the following EZB3881E message for the specific cause of the error.
In the event of an 804 abend, verify that the RIT exists and is accessible to NCPROUTE.

Module
NRPDUS

Procedure name
recv_hello

EZB3881E  Abend code: abend, Reason: reason

Explanation
An abend occurred during an attempt to load the Routing Information Table (RIT) into memory with the LOAD
macro. The abend and reason code returned are displayed. The session with the server is not established.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
See z/OS MVS System Codes for an explanation of the abend and reason codes, and further instructions.

Module
NRPDUS

Procedure name
recv_hello

EZB3882E A session will not be established

Explanation
An error occurred during an attempt to establish a session with a client, and the session cannot be established at this time.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the errors described in previous messages.

Module
NRPDUS

Procedure name
recv_hello

EZB3883E Client: client

Explanation
An error occurred during a transaction with the client. The client is the IP address of the client. This message is used to identify the failing client, and is followed by additional messages.

System action
NCPROUTE issues additional messages.

Operator response
None.

System programmer response
Examine the messages that follow for a description of the error.
Module
NRPDUS

Procedure name
recv_hello

EZB3884E The RIT ID field does not match the one in the Hello. RIT idfield, Hello: corrfield

Explanation
The ID field in the Routing Information Table (RIT) does not match the correlation field passed from the client NCP in the hello datagram. The mismatched ID and correlation fields are displayed. This indicates that the client is using a generation that is different from the one that built the RIT. A session is not established.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Generally, the RIT used by NCPROUTE should be the RIT that was generated during the generation of the current NCP load. In this case, an older RIT with the same name is probably being loaded. Locate and remove the old RIT. In some cases, multiple machines might want to own a single NCP and use the same NCP generation, but would require a unique RIT to run correctly. See the NCP documentation for instructions about how to set the correlation string in your generations and during NCP load.

Module
NRPDUS

Procedure name
recv_hello

EZB3885I Input parameters: parms

Explanation
This message lists the string of input parameters passed to NCPROUTE from tcpip.v3r1.SEZAINST(NCPROUTE) start proc JCL. The parameters could either be passed from the command line parameters or from the defined parameter list in the start proc JCL. If no parameters are specified, ‘*None.*’ will appear in the string.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
Module
NRMAIN

Procedure name
main

**EZB3886E** An error occurred while loading the RIT for client *client*.

**Explanation**
An internal error occurred. As a result of the error, the information in the Routing Information Table (RIT) is not available but an error condition was not indicated on return from the load. The new session ends.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

Module
NRPDUS

Procedure name
recv_hello

**EZB3887E** An internal error occurred, terminating session.

**Explanation**
Because a session was established, but is unusable, the new session ends and this message is displayed.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

Module
NRPDUS

Procedure name
recv_hello

**EZB3888E** An error occurred during interface initialization.
**Explanation**

An error occurred during processing of the Routing Information Table (RIT) interface list. Previous messages should describe the nature of the error. The new session ends.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Examine previous messages and follow recommendations for the error that occurred.

**Module**

NRPDUS

**Procedure name**

recv_hello

**EZB3889E**  The session will be terminated

**Explanation**

An error occurred that makes one of NCPROUTE’s sessions unusable. The session ends, and the client NCP goes into a reset state in three minutes. At that point, the NCP client attempts to reestablish a session.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Examine previous messages to determine what error occurred and correct the problem. If the error takes a while to correct, renaming the Routing Information Table (RIT) causes future attempts at establishing a session to be unsuccessful.

**Module**

NRPDUS

**Procedure name**

recv_hello

**EZB3890I**  Recv: status from client

**Explanation**

A client’s interface has changed state and the client has reported this by an interface status change PDU.
**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRPDUS

**Procedure name**
recv_status

**EZB3891I** Interface *ip_addr* is now status - *interface_name*

**Explanation**
The client’s interface name having an IP address of *ip_addr* has changed state and is currently set to either “up” or “down”.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
If the status change is not expected, examine the client NCP to determine the reason for the change. See the NCP documentation for more information.

**Module**
NRPDUS

**Procedure name**
recv_status

**EZB3894I** Transport from *client*: *count* bytes of RIP data.

**Explanation**
Client *client* has received a packet addressed to UDP port 520 and has forwarded this packet to NCPROUTE in a Transport PDU.

**System action**
NCPROUTE continues.

**Operator response**
None.
System programmer response
None.

Module
NRPDUS

Procedure name
recv_transport

EZB3895I  Transport from client: count bytes of SNMP data.

Explanation
The client has received a packet addressed to UDP port 161 and has forwarded this to NCPROUTE in a transport PDU.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRPDUS

Procedure name
recv_transport

EZB3896E  Transport for unsupported port port number bytes received and discarded

Explanation
The client NCP has sent NCPROUTE a Transport PDU containing a packet that was addressed to a port that NCPROUTE does not support. The transport is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRPDUS
**procedure name**
recv_transport

**EZB3897I**  NCPROUTE Server started

**Explanation**
NCPROUTE has completed initialization and is waiting for incoming packets from NCP client(s) to initiate sessions.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRMAIN

**procedure name**
main

**EZB3898I**  Recv: Inactive Interface List from client number interface(s) found:

**Explanation**
After a session is established, the client sends NCPROUTE a list containing the number of interfaces that are currently down. All other interfaces (listed in the Routing Information Table (RIT)) are assumed to be active. Following this message, the IP addresses and names of the inactive interfaces are displayed.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRPDUS

**procedure name**
recv_inactlist

**EZB3899I**  ipaddr - interface
**Explanation**
The IP address and name of the inactive interface are displayed.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRPDUS

**Procedure name**
recv_inactlist

---

**EZB3900W**  Unable to open NCPROUTE profile *profile*

**Explanation**
A profile data set was not specified or could not be opened. This message indicates which data set the open was attempted on.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Verify that the profile data set is defined in //DD:NCPRPROF of the NCPROUTE start proc JCL and is accessible by NCPROUTE. If the data set is sequential, ensure that the FREE=CLOSE parameter is specified. Do not specify this parameter if the data set is partitioned.

**Module**
NRMAIN

**Procedure name**
read_profile

---

**EZB3901W**  Ignoring route *destination*, supernetting not supported

**Explanation**
The route to *destination* received in the RIP packet happens to be a supernet type of route and is ignored since the supernetting feature is not supported by the NCP client. A supernet route is one where its subnet mask is less than the route’s network class mask.
**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Verify the NCP software level for supernetting support in the NCP client.

**Module**

NRINPUT

**Procedure name**

rip_input

---

**EZB3902W  Address is experimental, or has non-zero port**

**Explanation**

An incorrect IP address was encountered, which is either in an experimental address class, or is using an unusual port number (Routing Information Protocol (RIP) packets only). The address is being validated to make sure that a network user does not pretend to be a router to change the routing table of nearby routers (such as the client NCP). The address is not considered as a valid destination address for a route, and the route entry is discarded.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Identify the machine or user that generated the packet in question and correct the problem.

**Module**

NRAF

**Procedure name**

inet_checkhost

---

**EZB3903W  Invalid internet address**

**Explanation**

An IP address in an incoming route is determined to not be a member of any defined IP address class. The route is discarded.

**System action**

NCPROUTE continues.
Operator response
None.

System programmer response
Correct the router that originated this packet.

Module
NRAF

Procedure name
inet_checkhost

EZB3904W  A must-be-zero field is non-zero

Explanation
An IP address in an incoming route contains a nonzero value in a field that must be 0. The incoming route is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the router that originated this packet.

Module
NRAF

Procedure name
inet_checkhost

EZB3905S  function function client client unknown

Explanation
The function was unable to obtain a list of interfaces for the client with IP address client. If client is an actual client of NCPROUTE, the interface list has been lost, otherwise the function is using an incorrect address for its client. NCPROUTE indicates that the requested interface does not exist.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.
**EZB3906S**  NULL interface list for client client

**Explanation**
A list of interfaces for client client exists, but is marked empty. Dynamic routing requires at least one interface. The requested interface is indicated as nonexistent.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Verify that the client NCP generation had at least one interface that did not have the RIPMGD=NO keyword coded. If none are found, correct the configuration and regenerate the client. Verify that EZB3956I messages are issued which correspond with the Routing Information Protocol (RIP)-managed interfaces in the generation. If none are found, attempt to reload the Routing Information Table (RIT). Contact IBM software support services if the problem persists.

---

**EZB3908I**  Modify command is set for all clients

**Explanation**
NCPROUTE is configured to process MODIFY commands for all clients rather than for a specific client.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.
**Procedure name**

do_modify

**EZB3910I**  Modify command is set for client *client*

**Explanation**

NCPROUTE is configured to process MODIFY commands for a targeted client rather than for all clients.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

NRMAIN

**Procedure name**

do_modify

**EZB3911S**  Function *function* address family out of range. The address is *address*.

**Explanation**

One of NCPROUTE’s route entries has an unsupported address family. NCPROUTE cannot determine the network based on the address, therefore it cannot determine an interface that serves the logical network. The requested interface is indicated as nonexistent.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Noninternet addresses are not supported in this version of NCPROUTE. Either ignore the message, or have the router that originated this route stop sending noninternet routes to the client NCP. Look back through the output to find the last ADD or CHANGE for this destination to obtain the IP address of the router.

**Module**

NRIF

**Procedure name**

ifwithnet

**EZB3912I**  ifwithnet: compare with *interface*
Explanation
The route's network address is being compared with one of NCPROUTE's interface entries for a network number match.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRIF

Procedure name
if_ifwithnet

EZB3913I ifwithnet: remote interface ignored

Explanation
One of NCPROUTE's interface entries is for a remote interface, which is ignored during the search for a network number match.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRIF

Procedure name
if_ifwithnet

EZB3914S ifwithnet: interface has bad address family

Explanation
One of NCPROUTE's interface entries has an incorrect address family. NCPROUTE cannot determine the network based on the address. Therefore, it cannot find an interface that serves the logical network.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
Either ignore the message or correct the NCP client's interface that contains the incorrect address family. If the interface cannot be corrected, contact IBM software support services.

Module
NRIF

Procedure name
if_ifwithnet

EZB3915I  netmatch ipaddr1 and ipaddr2

Explanation
A network number match was found for the route's network address ipaddr1 with one of NCPROUTE's interface entries having a network address of ipaddr2.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRIF

Procedure name
if_ifwithnet

EZB3916I  Blocking route for destination

Explanation
From a packet received over a particular interface, the route for destination is being blocked from being added to NCPROUTE's routing table. This is the result of a RIP I/O filter option specified in the client's hlq.ETC.GATEWAYS data set. See z/OS Communications Server: IP Configuration Reference for more information.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
None.

Module
NRINPUT

Procedure name
rip_input

EZB3917I   NCPROUTE's internal type table for client client:

Explanation
NCPROUTE's internal IP routing or interface table is displayed for diagnosis.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTABLES, NRIF

Procedure name
dsp_rrtables, dsp_iftables

EZB3918I   Modify parameters processed; see SYSPRINT or SYSERR output for results

Explanation
The MODIFY command with specified parameters has been processed. For results, see the SYSPRINT or SYSERR output.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRMAIN
**Procedure name**
do_modify

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZB3921I</td>
<td>Tracing debug packets action timestamp</td>
</tr>
</tbody>
</table>

**Explanation**
Debug packets tracing is enabled. The packets are displayed in data format.

**System action**
NCPRoute continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRTRACE

**Procedure name**
modifydebuglevel

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZB3922S</td>
<td>inet_makeaddr: no session with client</td>
</tr>
</tbody>
</table>

**Explanation**
The function inet_makeaddr was unable to obtain a list of interfaces for the client with IP address client. If client is an actual client of NCPROUTE, the interface list has been lost; otherwise, the function is using an incorrect address for its client. A NULL IP address is created.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

**Module**
NRINET

**Procedure name**
inet_makeaddr

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZB3923S</td>
<td>inet_netof: no session with client</td>
</tr>
</tbody>
</table>
Explanation
The function inet_netof was unable to obtain a list of interfaces for the client with IP address client. If client is an actual client of NCPROUTE, the interface list has been lost; otherwise, the function is using an incorrect address for its client. Because a subnetmask cannot be located, a network number is returned.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRINET

Procedure name
inet_netof

EZB3924I  inet_netof: ignoring REMOTE interface

Explanation
One of NCPROUTE's interface entries is for a remote interface, which is ignored during the calculation for a (sub)network number from the interface's IP address.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRINET

Procedure name
inet_netof

EZB3925S  Inet_lnaof: no session with client

Explanation
The function inet_lnaof was unable to obtain a list of interfaces for the client with IP address client. If client is an actual client of NCPROUTE, the interface list has been lost; otherwise, the function is using an incorrect address for its client. Because subnetmask information is not available, the host part of the IP address is returned.
**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services. This might contain a subnet number if the interface is subnetted.

**Module**
NRINET

**Procedure name**
inet_lnaof

**EZB3926S**  
inet_rtflags: no session with *client*

**Explanation**
The function inet_rtflags was unable to obtain a list of interfaces for the client with IP address *client*. If *client* is an actual client of NCPROUTE, the interface list has been lost; otherwise, the function is using an incorrect address for its client. A determination is made to indicate either a network address or a host address assuming no subnetting has been done.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

**Module**
NRINET

**Procedure name**
inet_rtflags

**EZB3927E**  
Packet from unsupported address family (*family*), cmd (*command*)

**Explanation**
A packet was received from a non-Internet network. RIP traffic is only accepted from machines running IP. The incoming packet is discarded.

**System action**
NCPROUTE continues.
**Operator response**
None.

**System programmer response**
Locate the machine generating the RIP packet and have it stopped. These messages can also be ignored.

**Module**
NRINPUT

**Procedure name**
rip_input

**EZB3928E  RIP version 0 packet received from ipaddr**

**Explanation**
A Routing Information Protocol (RIP) Version 0 packet was received from the specified address. This RIP version is obsolete. The incoming packet is discarded.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Correct the router that is sending the Version 0 packets.

**Module**
NRINPUT

**Procedure name**
rip_input

**EZB3929I  Request: output routines removed**

**Explanation**
NCPROUTE received a Routing Information Protocol (RIP) request packet for a specific set of routes rather than for a set of all routes. (A request for all routes is indicated by an address family of 0 and infinite metric of 16.) Since NCPROUTE does not support handling requests for a specific set of routes, the output routines in this case have been removed to prevent a response.

**System action**
NCPROUTE continues.

**Operator response**
None.
System programmer response
None.

Module
NRINPUT

Procedure name
rip_input

EZB3930W Trace command from unknown router ipaddr

Explanation
A trace packet was received from a router that is either not directly connected to any of the client's Routing Information Protocol (RIP) managed interfaces, or is directly connected to an interface that is not capable of supporting RIP traffic. RIP requires that an interface be capable of supporting link-level broadcast traffic, be a point-to-point interface (such as NCST sessions), or have an active gateway defined. The incoming packet is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Determine where the packet originated from and correct the problem. Bridges can allow routers on logically disconnected networks to talk with each other even though the routers have no logical connection to each other's networks.

Module
NRINPUT

Procedure name
rip_input

EZB3931W Response from non-router ipaddr

Explanation
A Routing Information Protocol (RIP) response packet was received with an incorrect port number. All routers on a network must agree on the port number that is used to exchange routing information, and this port must be restricted so that other applications cannot generate routing updates. Either a router is configured using the wrong port number, or an application issues RIP routing updates. The RIP response is discarded.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
Reconfigure the router to use a correct port number or locate the application that is generating the updates and correct the problem.

Module
NRINPUT

Procedure name
rip_input

EZB3932W Invalid packet from passive interface interface

Explanation
A Routing Information Protocol (RIP) response packet was received from a passive interface. Passive interfaces receive routing updates from the client, but cannot produce routing updates. The packet is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Locate the router that is producing the bogus packet and correct the problem.

Module
NRINPUT

Procedure name
rip_input

EZB3933W Packet from unknown router ipaddress (reason)

Explanation
A Routing Information Protocol (RIP) response was received from a router which is not directly connected via a broadcast network, a point-to-point (NCST) network, or an active gateway as defined in a GATEWAYS PDS member. The description of the reason for this warning is one of the following:

Reason
Explanation
Interface in strange state
The network does not support broadcast or point-to-point transmission.

Iflookup failed
Not directly connected.

Link is PASSIVE!
Cannot update.

The packet is discarded.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
Locate the router that produced the packet and correct the problem.

Module
NRINPUT

Procedure name
rip_input

EZB3937W  New route is in unsupported address family. client = client, route from ipaddr, new family = family

Explanation
An incoming router from another router is in an address family that is not supported by NCPROUTE. Currently, only Internet addresses are supported. The client is the IP address of the client NCP, the ip addr is the IP address of the router that originated the route that is not valid, and family is the address family that is not supported by NCPROUTE. The route that is not valid is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Locate the router that produced the route and correct the problem.

Module
NRINPUT

Procedure name
rip_input

EZB3939E  Illegal address hostaddr in route from ipaddr

Explanation
An IP address that is not valid was received in an update from router ipaddr. A previous message indicates the nature of the problem with the address. The route that is not valid is discarded.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
Locate the router that originated the route and correct the problem.

Module
NRINPUT

Procedure name
rip_input

EZB3940W  Bad metric (metric) in route to destination from router ipaddr

Explanation
A route was received from ipaddr that contained a metric that was not in the range 1 - 16. The route is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Locate the router ip address and correct the problem.

Module
NRINPUT

Procedure name
rip_input

EZB3941I  Adjusting large metric (metric) to infinity

Explanation
From the routing updates, a route contained a metric exceeding the maximum supported metric of 16. The metric is changed to infinite metric of 16, to indicate that the destination route is network unreachable.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
**EZB3943I  Send dynamic update**

**Explanation**
Changes have occurred and an update has not been sent recently. Enough time has passed since the last update was sent so that it is safe to transmit again without risking an update storm. Also, no dynamic update was pending; otherwise, NCPROUTE would wait until that update occurred before sending the update.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRINPUT

**Procedure name**
rip_input

---

**EZB3944I  Delay dynamic update**

**Explanation**
Changes have occurred in the network topology, but an update was recently made to adjacent routers. When the last update was made, a random delay time, 2-5 seconds from that point, was determined. This update is scheduled to occur at that time.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRINPUT

**Procedure name**
rip_input

---

**EZB3945I  Inhibit dynamic update for seconds usec**
Explanation
A dynamic update has just been sent. Another dynamic update is prevented from occurring for the number of microseconds indicated in the message. A random time is chosen, 2-5 seconds from this time, and if another update is needed later, it will be delayed until this random time has passed.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRINPUT

Procedure name
rip_input

EZB3946S toall (STUB) -- this shouldn't be used anymore

Explanation
The routine (toall) called by NCPROUTE is no longer supported.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NROUTPUT

Procedure name
toall

EZB3947S toall: client client unknown

Explanation
Information about the client is not available, but NCPROUTE is attempting to send output over all of the client's interfaces. This is an internal error. No output is sent to the client.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
None.

Module
NROUTPUT

Procedure name
toall_ifs

EZB3948I Interface interface not up

Explanation
The specified interface is detected to be inactive. No route will be added unless the interface is (re)activated.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSTART

Procedure name
addrouteforif

EZB3949I Interface interface is passive

Explanation
The interface is in a passive state, meaning that RIP traffic is disabled for the interface. Routing updates will not be broadcast to the interface and incoming routing updates are ignored. This may be the result of a RIP I/O filter option specified in the client's hlq.ETC.GATEWAYS data set. See z/OS Communications Server: IP Configuration Reference for more information.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
Module
NROUTPUT

Procedure name
toall

EZB3950I  toall: requested to skip interface interface

Explanation
The interface interface is skipped because the interface has already received notification of a routing change. If a broadcast of the routing table has not been sent recently and a routing change has occurred, a dynamic routing update will be broadcasted to other interfaces to inform adjacent routers of the routing change.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NROUTPUT

Procedure name
toall

EZB3951I  client client: supply destaddress -> port via interfacename

Explanation
NCPROUTE is directing the client to transmit a Routing Information Protocol (RIP) datagram to the destination address. This can be either a broadcast address or a host address. If the port is zero, the RouteD port will be used from hlq.ETC.SERVICES; otherwise, the datagram will be transmitted to the specified port. NCPROUTE will ask the client to use the specified interface name when sending the datagram.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NROUTPUT
**EZB3952E**  Unknown interface name *(interface)* and/or address *(ip_addr)*

**Explanation**
In the line entry for the client's *hlq.ETC.GATEWAYS* data set, the options definition contains interface information that is not in NCPROUTE's interface tables. Most likely, the interface name is misspelled or the interface's IP address is specified incorrectly. Although other options may be processed normally, the invalid option is ignored.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Correct the client's *hlq.ETC.GATEWAYS* data set. See *z/OS Communications Server: IP Configuration Reference* for more information.

---

**EZB3953S**  NULL base

**Explanation**
NCPROUTE was unable to locate a base hash table for the client. Every active client should have this table, so its absence indicates an internal error. No output occurs.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

---

**EZB3954W**  Invalid metric, changing to one
Explanation
In the line entry for the NCP client's hlq.ETC.GATEWAYS data set, the metric has an incorrect value. The metric is changed to 1.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the NCP client's hlq.ETC.GATEWAYS data set.

Module
NRSTART

Procedure name
gateways

EZB3955S Function function: out of memory

Explanation
NCPROUTE was unable to allocate memory because no more storage is available in the region. The following describe functional errors:

Function Error Description
ifinit No storage available to add a client interface entry.
nradd No storage available to add a route table entry. Each route table entry requires 64 bytes.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Increase the region size in the startup procedure.

Module
various

Procedure name
various

EZB3956I Processing interface interface
Explanation
The indicated interface was found in the Routing Information Table (RIT) for the new client and is being added to NCPROUTE's tables.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSTART

Procedure name
ifinit

EZB3957E    Modify command ignored, invalid parm(s): parms

Explanation
Invalid parameters were passed to NCPROUTE from a MODIFY command.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the parameters from the MODIFY command.

Module
NRMAIN

Procedure name

EZB3958E    Modify command ignored, client 'c=' not specified

Explanation
The parameters specified in the MODIFY command requires the target client specification.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
Specify the target client in the "c=" parameter of the MODIFY command.

Module
NRMAIN

Procedure name
parse_parms

EZB3959I   point-to-point interface, using addrtype

Explanation
The new interface is point-to-point, and the destination address type, addrtype, can either be dstaddr or broadaddr. The address type is determined by the interface definition in the NCP generation. The destination address may be coded such that the network or subnetwork directed broadcast address is used or that the unicast or host address representing the other end of the point-to-point link is used. This address is used for sending routing information over an interface to the destination router or host. The exception is when the interface is multicast-capable; the multicast address is used.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSTART

Procedure name
addrouteforif

EZB3960I   This interface is not point-to-point dstaddr

Explanation
The new interface is not point-to-point. A route is being added based on the interface definition, and the network or subnetwork route will be used as the route destination.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
None.

Module
NRSTART

Procedure name
addrouteforif

EZB3961I  not an internal interface

Explanation
The new interface does not appear to be associated with a real device, this interface is most likely a pseudo-interface created because of an external route in the GATEWAYS member for this client. No route is created for this interface.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSTART

Procedure name
addrouteforif

EZB3962I  Adding type route for interface

Explanation
The route using the network, subnetwork or destination type is being added to the interface in NCPROUTE's routing table.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSTART
**Procedure name**
addrouteforif

**EZB3963I**  Re-installing interface *interface*

**Explanation**
The previously deleted interface is being re-installed since traffic has been detected over this interface.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSTART

**Procedure name**
addrouteforif

**EZB3965I**  Route seq: *dest* via gateway metric *metric*, supnetmask *mask*

**Explanation**
A route is being added based on an IPROUTE statement coded during NCP generation. The *seq* is the position in the route table, *dest* is the destination IP address, *gateway* is the nexthop for the route, and *metric* is the cost of using this route, and *mask* is the subnet mask for the route.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSTART

**Procedure name**
bld_rtbl

**EZB3967W**  Unable to open a GATEWAYS dataset for client *client*. Attempt to open *dataset*
**Explanation**
A gateways data set was not specified, or could not be opened for the specified client. This message indicates which data set the open was attempted on.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
If a gateways data set is being used, verify that the PDS member name is coded correctly in the *hlq.PROFILE.TCPIP* data set. Verify that the gateways member has the same name as the client NCP name. Verify that NCPROUTE has access to this data set member. Verify that NCPROUTE attempts to open the correct data set member. Correct the PROFILE and rename the PDS member, if needed. Contact IBM software support services if the problem persists.

**Module**
NRSTART

**Procedure name**
gateways

**EZB3968I**  
Start of GATEWAYS processing:

**Explanation**
The gateways data set member is about to be processed. Messages about data set processing might follow.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSTART

**Procedure name**
gateways

**EZB3969E**  
statement

**Explanation**
The statement shown is in error. A message describing the error will follow.
EZB3970E  Invalid gateway address “gateway”

Explanation
This statement’s gateway is neither a resolvable name nor a valid dot-notation IP address. The statement is ignored.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSTART

EZB3971E  Zero metrics not allowed, changing to one

Explanation
This statement has a 0 metric, which is not valid, Metrics must be between 1 and 15. The metric is changed to 1.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
System programmer response
Correct the statement in the gateway member.

Module
NRSTART

Procedure name
gateways

EZB3972E Gateway type “passive” not valid for active gateway

Explanation
An active gateway entry is qualified as a passive route. Active is the only valid route type for this definition. The statement is ignored.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the statement in the gateway member.

Module
NRSTART

Procedure name
gateways

EZB3973I Opening GATEWAYS dataset for client client. dataset

Explanation
The gateways data set is being opened for the specified client. Entries in the data set are read in for input.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSTART
**EZB3974E**  
First two keywords must be 'active' for active gateway.

**Explanation**  
A GATEWAYS entry for the gateway definition contains an active route type, but the first two keywords are not defined as active. These keywords are required for an active gateway definition. If this entry is not for an active gateway, correct the route type. Active gateway entries identify only a router and have no destination information. The GATEWAYS entry is ignored.

**System action**  
NCPROUTE continues.

**Operator response**  
None.

**System programmer response**  
Correct the GATEWAYS entry.

**Module**  
NRSTART

---

**EZB3975E**  
Invalid gateway type type

**Explanation**  
This statement does not end with a valid route type, either “active”, “passive” or “external”. The statement is ignored.

**System action**  
NCPROUTE continues.

**Operator response**  
None.

**System programmer response**  
Correct the statement in the gateways member.

**Module**  
NRSTART

---

**EZB3976S**  
nr_exact_find: no session with client
**Explanation**
A route to a destination from a nonexistent client is being requested. This is an internal error. The route is reported as not existing.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

**Module**
NRTABLES

**Procedure name**
nr_exact_find

**EZB3977S**  

nr_exact_find: no network hash table for client *client*

**Explanation**
Client *client* does not appear to have a required network hash table. This is an internal error.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

**Module**
NRTABLES

**Procedure name**
nr_exact_find

**EZB3978S**  

nr_kernel_find: no host hash table for client *client*

**Explanation**
Client *client* appears to be missing a required host hash table. This is an internal error.

**System action**
NCPROUTE continues.
Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRTABLES

Procedure name
nr_kernel_find

EZB3979S nr_kernel_find: no network hash table for client client

Explanation
The session with client has been incorrectly initialized. A required network hash table is missing.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRTABLES

Procedure name
nr_kernel_find

EZB3980E nradd: invalid address family

Explanation
An attempt is being made to add a route which is to a destination in an unsupported address family. Currently, NCPROUTE only supports routes to Internet addresses. The add is not performed.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Identify the machine that generated the route and correct the problem.
Module
NRTABLES

Procedure name
nradd

EZB3981S  nradd: no host hash table for client client

Explanation
A host route is being added, but the client’s host hash table cannot be located. This is an internal error. The route is not added.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRTABLES

Procedure name
nradd

EZB3982S  nradd: no network hash table for client client

Explanation
A network or subnetwork route is being added, but the required network hash table for client cannot be located. This is an internal error. The route is not added.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRTABLES

Procedure name
nradd

EZB3983E  Modify command ignored, type trace levels exceeded
Explanation
An incorrect number of trace levels (-t's) were passed from a MODIFY command.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Specify a correct number of -t's in the MODIFY command line parameters.

Module
NRMAIN

Procedure name
ezb3984E  error adding route to host/net destination through gateway

Explanation
The NCP client failed to add a route.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Examine previous error messages to determine the nature of nr_do_add’s error.

Module
NRTABLES

Procedure name
nradd

ezb3989I  changing/deleting route to interface interface (timed out?)

Explanation
Either traffic to a local interface is being routed through a remote gateway, or the metric on the interface has increased to infinity. Usually this indicates that the interface has timed out and is considered down.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
If the interface is being assigned a new gateway, examine the contents of the last packet sent from the gateway. The gateway’s address is found in an upcoming message. Look for a display of the local interface from the remote gateway. Correct the remote gateway if needed. If the interface is timing out, verify that no transmissions have been received over this interface for the last 3 minutes. If no transmissions are found, this is correct behavior for NCPROUTE, and the physical lines or remote gateways should be examined to determine the cause of the problem.

Module
NRTABLES

Procedure name
nrchange

EZB3993E could not delete route to type destination via router

Explanation
The route to destination is being deleted (possibly because the route is changing, in which case an add will follow) and nr_do_delete returned an error. type is one of Host, Network, Subnet, or (bad_type). (bad_type) indicates an internal error and should be reported. Additional errors might follow because of the unsuccessful attempt to remove the route. These should be ignored until this problem is resolved. The route remains.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Examine previous messages to determine the error that occurred in nr_do_delete, contact IBM software support services with this information.

Module
NRTABLES

Procedure name
nrchange

EZB3994E Error adding route

Explanation
An error occurred while attempting to update a route. The route is not readded and the net effect is that the route is deleted.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
Examine previous messages to determine the error that occurred in nr_do_add and contact IBM software support services with this information.

Module
NRTABLES

Procedure name
nrchange

EZB3996I  deleting route to interface interface? (timed out?)

Explanation
A route to the indicated interface is being deleted and the metric is less than infinity.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTABLES

Procedure name
nrdelete

EZB3997E  Error deleting route

Explanation
The NCP client failed to add a route. Additional errors might follow this one. Because the route was not deleted, these should be ignored until this problem is resolved.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Examine previous error messages to determine the error that occurred in nr_do_delete and contact IBM software support services with this information.
Module
NRTABLES

Procedure name
nrdelete

EZB3998S nr_client_init: no session established with client

Explanation
A session with the indicated client does not appear to have been initialized correctly.

System action
NCROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRTABLES

Procedure name
nr_client_init

EZB3999I Establishing session with client client

Explanation
A session is being established with a NCP client. This is required before PDUs can be processed from the client.

System action
NCROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTABLES

Procedure name
nr_establish_session
Chapter 6. EZB4xxxx messages

**EZB4000I**  Terminating session with client *client*

**Explanation**
The session with the indicated client ends. This is an orderly shutdown, and occurs in the following cases: upon receipt of a hello from an active client, if the load of the Routing Information Table (RIT) is unsuccessful, or if an error occurs when processing the RIT. Other messages occur earlier that explain which case caused the session to be brought down.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None. required. If the session is brought down in response to an error, examine previous messages to determine the nature of the error, and correct the problem.

**Module**
NRTABLES

**Procedure name**
nr_terminate_session

**EZB4001S**  Terminate Session: no session with client *client*

**Explanation**
The NCP client failed to add a route. unacceptable client handle. The client cannot be located in the session table. No session ends.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Contact IBM software support services.

**Module**
NRTABLES

**Procedure name**
nr_terminate_session
EZB4002I  Chain chain_number

Explanation
The session table is being displayed. Following this message is the contents of chain chain_number.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTABLES

Procedure name
dsp_sessions

EZB4003I  Hosts:

Explanation
The session table is being displayed. Following this message is the contents of the host route table for the current session.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTABLES

Procedure name
dsp_sessions

EZB4004I  Networks:

Explanation
The session table is being displayed. Following this message is the contents of the network route table for the current session.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTABLES

Procedure name
dsp_sessions

EZB4005I  No Entries

Explanation
This particular route table is empty.

System action
NCPROUTE continues

Operator response
None.

System programmer response
None.

Module
NRTABLES

Procedure name
dsp_rtbl

EZB4006I  Subchain subchain_number

Explanation
A routing table is being displayed. The following messages contain the contents of the routing table chain subchain_number.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
None.

Module
NRTABLES

Procedure name
dsp_rtbl

EZB4007I Entry empty

Explanation
The route table subchain currently being displayed is empty.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTABLES

Procedure name
dsp_rtbl

EZB4008I Entry: entry

Explanation
The route table subchain currently being displayed contains an entry.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTABLES
Procedure name
dsp_rtbl

EZB4009I  client client: timer_value minute timer expired for route to destination

Explanation
No Routing Information Protocol (RIP) packets have been received from the NCP client to the destination destination in the last timer_value minutes. It is assumed that the destination route is no longer active. Depending upon the timer_value, one of the following actions is taken:

Value  Action
3   The route will have its metric changed to infinity for the next 2 minutes. The metric change is necessary to alert adjacent routers that the route to this destination is unreachable. If NCPROUTE receives any RIP packets for the NCP client from the destination router during the 2 minute time interval, NCPROUTE will restore the route by changing the metric to a valid one based upon the received RIP packet. The route will be deleted from the NCP client's routing table.
5   The route will be deleted from NCPROUTE's routing table for the NCP client.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
These actions require the following responses:
Value  Action
3   If the route was broadcasted for a while, and then suddenly stopped, look for an adapter problem, or a problem with the physical line.
5   Examine the NCPROUTE trace output and determine when the broadcasting has stopped for the route to the destination router. If the route to destination was only broadcast once in response to the NCPROUTE's request for full route tables for the NCP client, then the problem may be with the way RIP packets are broadcasted.

In these cases, determine if NCPROUTE is receiving the transport PDUs containing the RIP packets from the NCP client by obtaining a MORETRACE IPUP trace and look for "discarding broadcast" packet messages. This trace can confirm the lack of traffic on the interface used for the transport PDUs forwarded by the NCP client. If this is not the case, then determine if NCPROUTE is receiving the RIP packets over the NCP client's routing interface by obtain- ing a NCP line trace. See the NCP documentation for instructions on obtaining a line trace.

Module
NRTIMER

Procedure name
client_timer

EZB4010I  client client: 30 second timer expired (broadcast)
**Explanation**

Every 30 seconds, a timer expires that indicates that a client NCP must broadcast its routing tables to adjacent routers. NCPROUTE will build Routing Information Protocol (RIP) response packets for each of the client's interfaces and send them to the client in a transport PDU. The client transmit them on the specified interface.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

NRTIMER

**Procedure name**

client_timer

---

**EZB4011E**  Invalid subnetwork mask *mask*

**Explanation**

In the line entry for the NCP client's gateways data set, the gateway definition has a subnetmask that is not valid. The subnetmask must be a resolvable subnetmask name, or a bit mask in dotted-decimal notation. The gateway entry is ignored.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Correct the NCP client's gateways data set.

**Module**

NRSTART

**Procedure name**

ParseOptions

---

**EZB4012E**  Trace buffers not initialized for interface *interface* on client *client*

**Explanation**

One or both trace buffers could not be obtained for the specified interface during initialization. This is because of a lack of free storage in the region. Minimal tracing is performed.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
Increase the region size for NCPROUTE and restart NCPROUTE.

Module
NRTRACE

Procedure name
traceinit

**EZB4013I**  Tracing action for client client

Explanation
Tracing is enabled or disabled for a client NCP.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTRACE

Procedure name
various

**EZB4014W**  Unknown RIP rip_control control value (value)

Explanation
The RIP control values (supply or receive), specified on the RIP_SUPPLY_CONTROL or RIP_RECEIVE_CONTROL entry in the NCPROUTE profile data set, or specified on the options statement entry for a client for all interfaces or for an interface, contains an incorrect value. In case of incorrect values, NCPROUTE will default the RIP supply control to 'RIP1' and receive control to 'ANY' for the NCP client.

System action
NCPROUTE continues.

Operator response
None.
**System programmer response**
Correct the NCPROUTE profile or the NCP client’s gateways data set by specifying a supported supply or receive control value.

**Module**
NRMAIN, NRSTART

**Procedure name**
read_profile, ParseOptions

**EZB4015I**
Client tracing actions started

**Explanation**
The current tracing level has been advanced to the "actions" level, which causes messages to be issued for actions, such as adding, changing, or deleting routes. Additional messages for actions, such as waiting for incoming packets and dynamic updates, are also issued.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRTRACE

**Procedure name**
trace_start

**EZB4016I**
Client tracing packets started

**Explanation**
The current tracing level is advanced to the "packets" level, which displays the types of packets sent and received in addition to the output displayed at the "actions" level.

**System action**
NCPROUTE continues. Correct the hlq.NSMAIN.DATA data set to include a server name for the indicated address, or enter the NSLOOKUP command, using the server_name and server_address parameters to specify the default Domain Name Server. For more information about the hlq.NSMAIN.DATA data set, see the z/OS Communications Server: IP Configuration Reference.

**System programmer response**
None.

**Module**
NRTRACE
**Procedure name**
trace_start

**EZB4017I**  Client tracing history started

**Explanation**
The current tracing level is advanced to the "history" level, which displays history tracing data for each line in addition to output displayed at the "packets" level. The history tracing data is displayed whenever an interface becomes inactive. It shows the latest traces of actions, packets and packet contents before the interface became inactive.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRTRACE

**Procedure name**
trace_start

**EZB4018I**  Client tracing packet contents started

**Explanation**
The current tracing level is advanced to the "packet contents" level, which displays the contents of packets sent or received in addition to output displayed at lower tracing levels. Additional messages, such as requests for full routing tables and unknown address family in routing information, are also issued.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRTRACE

**Procedure name**
trace_start

**EZB4029I**  timestamp:

**EZB4029I**  timestamp:
Explanation
A full time stamp is issued showing the date and time so that traces that exceed one calendar day can be interpreted correctly.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTRACE

Procedure name
trace

EZB4030I  action destination destination router router, metric metric, flags flags

Explanation
The route to the destination is being added, deleted, or changed depending on the action. The following action values are allowed:

Value  Explanation
ADD     The route to the destination is being added through the router at the specified metric.
CHANGE FROM  The route to the destination is being changed and the current values are displayed.
CHANGE TO   The route to the destination is being changed and the new values are displayed.
DELETE    The route to the destination is being deleted.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTRACE
**EJB4032E**  Gateway address ‘address’ not an routing interface in network

**Explanation**
The indicated passive gateway, defined in the NCP client's hlq.ETC.GATEWAYS data set, referenced an unknown routing interface based upon the gateway address. The passive route definition is ignored.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Correct the client's hlq.ETC.GATEWAYS data set. Verify that the gateway address is correct and a valid routing interface is defined. See z/OS Communications Server: IP Configuration Reference for more information about NCP Host Interface Definition.

**Module**
NRSTART

**Procedure name**
gateways

**EJB4036I**  CHANGE metric destination destination, router router, from old metric to new metric

**Explanation**
The metric for the route to the destination is being changed from the old metric to the new metric. This is always based on a Routing Information Protocol (RIP) packet from the router, which updates a previous route through the router.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRTRACE

**Procedure name**
tracenewmetric

**EJB4038I**  *** Packet history for interface interface ***
Explanation
Tracing is set at the history level, and the history trace data for the inactivated interface *interface* is displayed.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTRACE

Procedure name
dumpif

EZB4039I  *** End packet history ***

Explanation
Tracing is set at the history level, and this message ends the history trace data for the deactivated interface.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTRACE

Procedure name
dumpif

EZB4043I  *direction: no packets*

Explanation
Either the input or output trace buffer, depending on the direction value, is empty.

System action
NCPROUTE continues.
**Operator response**
None.

**System programmer response**
None.

**Module**
NRTRACE

**Procedure name**
dumptrace

**EZB4044I**  
*direction trace:*

**Explanation**
Tracing is currently at the packet history level. Either the input or output trace buffer is about to be displayed depending on the direction value.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRTRACE

**Procedure name**
dumptrace

**EZB4045I**  
*RIPcmd direction router -> port timestamp p*

**Explanation**
A Routing Information Protocol (RIP) datagram has been received, or is about to be sent out of NCPROUTE depending on the direction value. A value of “to” indicates an outbound datagram, while a value of “from” indicates an inbound datagram. The type of the datagram is indicated by *RIPcmd*, and can be either RESPONSE or REQUEST. Responses are displays of routes from one router to another, requests are requests for individual routes or full tables. The value of *port* is either the port number which the packet came in on, the value 0, which indicates that the datagram will be sent to the default *RouteD* port as described in ETC.SERVICES, or in the case of outbound datagrams with a nonzero value the port which the datagram will be sent to.

**System action**
NCPROUTE continues.

**Operator response**
None.
System programmer response
None.

Module
NRTRACE

Procedure name
dumppacket

EZB4046S     Bad cmd hex direction router -> port timestamp size=size cp=directino packet=addr
time=timestamp

Explanation
A malformed packet has been encountered during trace. The source of the packet is either the router, if the value of direction is “from”, or NCPROUTE if the value of direction is “to”.

System action
NCPROUTE continues

Operator response
None.

System programmer response
If the source of the packet is NCPROUTE, contact IBM software support services. Otherwise locate the router and take corrective action.

Module
NRTRACE

Procedure name
dumppacket Routing Information Protocol (RIP)

EZB4048E     (truncated record, len len)

Explanation
A Routing Information Protocol (RIP) datagram was received that did not end on a route boundary. Either the packet was too large, or it has been truncated.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Verify that the packet received was built correctly and can be processed with other routers. Contact IBM software support services if the problem appears to be with NCPROUTE.
Module
NRTRACE

Procedure name
dumppacket

EIZB4049I destination destination metric metric

Explanation
A route to the destination at the indicated metric is being displayed from NCPROUTE or an adjacent router depending on the contents of the last EZB4045I message. The metrics displayed to NCPROUTE by other routers do not have the interface metrics added to them, this is done when the route is added to NCPROUTE’s tables. Likewise, NCPROUTE displays routes at a metric which does not include the metric of the interface on which the route is being displayed. A route is never displayed on the interface from which it was received.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTRACE

Procedure name
dumppacket

EIZB4050I (request for full tables)

Explanation
A request for a complete routing table is being sent, or has been received depending on the contents of the last EZB4045I message. This message is sent during session initialization by NCPROUTE. In addition to foreign routers making this request, application programs can ask a client for its tables by sending a Routing Information Protocol (RIP) request for a route with an address family of 0 and a metric of 16 to a client NCP at the current RouteD port (by default, 520).

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
**Module**
NRTRACE

**Procedure name**
dumppacket

**EZB4051E** unknown address family *family* metric *metric*

**Explanation**
A route in an unacceptable address family has been received, or is about to be sent depending on the contents of the last EZB4045I message. Currently, only AF_INET is supported, although AF_UNSPEC is allowed when making requests for full tables.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
If the route is being sent from NCPROUTE, contact IBM software support services. Otherwise locate the router listed in the last EZB4045I message and correct it so that noninternet routes are not sent to the client NCP.

**Module**
NRTRACE

**Procedure name**
dumppacket

**EZB4052I** traceon file = *dataset*

**Explanation**
A TRACEON packet was received. Tracing is requested to go to the specified data set.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.
EZB4055I  attempting to (re)start SNMP connection

Explanation
NCPROUTE is attempting to establish a connection with the SNMP agent specified in the profile. This occurs during startup, and at any time when an SNMP request occurs and no connection exists with the agent.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
init_snmp

EZB4056W  no response from agent on host

Explanation
NCPROUTE was unable to establish a connection with the SNMP agent on host host.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Verify that an agent is running on host, and that IP routes exist between the local host and host. When tracing is activated on the SNMP agent, DPI requests should be seen coming from NCPROUTE.

Module
NRSNMP

Procedure name
init_snmp

EZB4057W  Start the agent before issuing SNMP queries.

Explanation
A reminder that the SNMP agent should be running before NCPROUTE will be able to process SNMP queries for its clients.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
init_snmp

EZB4058E  An error occurred while opening the SNMP socket:

Explanation
An error occurred while attempting to open a socket for communicating with the SNMP agent. A more detailed
tcpenv() library message follows.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the problem as indicated by the error in the detailed tcppenv() library message. See the z/OS XL C/C++
Runtime Library Reference for more information about socket() function errors.

Module
NRSNMP

Procedure name
init_snmp

EZB4059I  Connecting to agent agent on DPI port port

Explanation
NCPROUTE is attempting to connect to the SNMP agent using the DPI port.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
None.

Module
NRSNMP

Procedure name
init_snmp

EZB4060E  An error occurred while connecting to the SNMP agent:

Explanation
NCPROUTE was unsuccessful in connecting to the SNMP agent. A more specific error message will follow.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services

Module
NRSNMP

Procedure name
init_snmp

EZB4061E  A connection is required for processing SNMP requests.

Explanation
A reminder that the SNMP agent socket must be connected before SNMP queries are processed.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP
Procedure name
init_snmp

EZB4062I   SNMP DPI connection established

Explanation
The connection with the SNMP agent is established and SNMP queries will now be processed as they arrive.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
init_snmp

EZB4063E   Unable to register root with SNMP agent.

Explanation
NCPROUTE was unable to register the MIB extension root with the agent. An I/O error occurred while sending the registration to the agent. The connection to the agent is closed and will be reopened for the next query.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Verify that the agent is operating correctly and that the agent's host is currently reachable from the local host.

Module
NRSNMP

Procedure name
init_snmp

EZB4064I   root registered with SNMP agent

Explanation
The MIB extension root has been registered with the SNMP agent.
**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSNMP

**Procedure name**
init_snmp

---

**EZB4065E**  Unable to establish a session with the SNMP agent.

**Explanation**
An attempt to reestablish a connection with the SNMP agent has failed.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Verify that the SNMP agent is running. Verify that the SNMP statements in the NCPROUTE.PROFILE are correct, and verify that the host running the SNMP agent can be reached from the local host.

**Module**
NRSNMP

**Procedure name**
snmp_input

---

**EZB4066E**  An incoming SNMP packet was discarded (noagent)

**Explanation**
Because no connection exists with the SNMP agent, the SNMP packet that is being processed is discarded.

**System action**
NCPROUTE continues.

**Operator response**
None.
System programmer response
None.

Module
NRSNMP

Procedure name
snmp_input

EZB4067E  Error while forwarding SNMP packet to agent:

Explanation
NCPROUTE was unable to forward the SNMP request up to the SNMP agent. A more specific error message follows. The SNMP packet is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Verify that the SNMP agent is running and that the host running the agent is reachable from the local host.

Module
NRSNMP

Procedure name
snmp_input

EZB4068I  SNMP response from local agent

Explanation
The SNMP agent has responded to a forwarded request.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP
Procedure name
snmp_response

EZX4069E    Error while sending SNMP reply to client client

Explanation
NCPROUTE was unable to send the response to a SNMP query back to the client. The packet is discarded.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Examine the following error message, and take whatever corrective action is recommended.

Module
NRSNMP

Procedure name
snmp_response

EZX4070S    Couldn't parse incoming DPI packet! (dropping) This suggests a problem with the SNMP agent.

Explanation
NCPROUTE could not parse an incoming DPI packet from the SNMP agent. The packet is damaged or has not been built correctly. Either the SNMP agent or the SNMP dpi library is in error. Also, they may be at different maintenance levels.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRSNMP

Procedure name
dpi_in

EZX4071I    rip_control rip_version packets on interface interface not allowed
Explanation
A RIP Version 1 or Version 2 packet is ignored depending upon the settings of the RIP supply or receive controls specified in the NCP client's gateways data set for an interface or in the NCPROUTE profile data set. If there are no RIP control settings for an interface, NCPROUTE will use the one from the profile settings.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRINPUT, NROUTPUT

**Procedure name**
rip_input, supply

---

**EZB4072I**  SNMP: DPI GET request (oid) received.

**Explanation**
An SNMP DPI get request for the variable specified by the ASN.1 object identifier was received. The object identifier is a registered MIB extension, and if you remove the root of the object identifier (based on the contents of EZB4064I) you will have a client identifier followed by the original object identifier.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSNMP

**Procedure name**
dpi_in

---

**EZB4073I**  SNMP sub-agent: DPI GET NEXT request (oid) received.

**Explanation**
An SNMP DPI get next request for the specified object identifier has been received.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
dpi_in

EZB4074I  SNMP sub-agent: DPI SET request received

Explanation
An SNMP DPI set request was received. SNMP set commands are not supported in this release.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
dpi_in

EZB4075S  Unexpected SNMP query type type; SNMP support appears incomplete in NCPROUTE.

Explanation
An unexpected SNMP DPI packet type has been received from the SNMP agent. The packet has already been validated by the DPI library routines, so this indicates an unsupported valid DPI packet type. An error is returned to the originator of the SNMP packet.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
Contact IBM software support services.

Module
NRSNMP

Procedure name
dpi_in

EZB4076I RIP2 packet from router router not authorized

Explanation
A RIP Version 2 packet, received from router, is ignored as a result of a authentication key mismatch. Authentication is enabled for RIP Version 2 packets according to the interface options in the NCP client's gateways data set or in the NCPROUTE profile data set. If there are no interface settings, NCPROUTE will use the one from the profile settings.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRINPUT

Procedure name
rip_input

EZB4077S SNMP sub-agent: received DPI request outside tree

Explanation
The SNMP agent has requested information or action on a MIB variable that is outside of the tree that NCPROUTE manages. This indicates an error in the agent code, because NCPROUTE did not register this variable. NO_SUCH_NAME is returned.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Compare the object identifier for the current DPI packet with the region of the MIB tree that NCPROUTE manages. The object identifier can be found in the last EZB4072I, EZB4073I, or EZB4074I message issued. The registered region can be found in the last EZB4064I message issued.
Module
NRSNMP

Procedure name
dpi_in

EZB4078I   SNMP sub-agent: received invalid DPI request outside tree

Explanation
The DPI packet from the agent refers to an OID which is not present in the managed tree.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
dpi_in

EZB4079I   iproutedest.instance

Explanation
A request has been made for the destination of the specified route. Each route in the client’s table is numbered starting with 0.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
snmp_lookup

EZB4080I   iprouteifindex.destination
Explanation
A request has been made for the interface index associated with the route to destination for the client which forwarded this request to NCPROUTE.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
snmp_lookup

EZB4081I iproutemetric1. destination

Explanation
A request has been made for the metric associated with the route to the destination for the client that forwarded this request to NCPROUTE.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
snmp_lookup

EZB4082I iproutemetric(2-4). destination (unsupported)

Explanation
A request was made for one of the alternate route metrics that is not used under RIP. If a route to the specified destination exists, a value of ‘-1’ is returned; otherwise, NO_SUCH_NAME will be returned.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
snmp_lookup

EZB4083I    iproutenexthop.instance

Explanation
A request was made for the next hop for the specified route. Routes are specified by the instance number, which is the ordinal position in NCPROUTE's tables. If the specified instance exists, an IP address will be returned; otherwise, NO_SUCH_NAME will be returned.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
snmp_lookup

EZB4084I    iproutetype.instance

Explanation
A request was made for the route type for the specified route. If the specified instance exists, one of three values will be returned depending on the route state:

<table>
<thead>
<tr>
<th>Value</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>The route is to a directly connected destination.</td>
</tr>
<tr>
<td>Invalid</td>
<td>The route has an infinite metric.</td>
</tr>
<tr>
<td>Remote</td>
<td>The route is to an indirectly connected destination requiring one or more routers to reach.</td>
</tr>
</tbody>
</table>
**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSNMP

**Procedure name**
snmp_lookup

---

**EZB4085I  iprouteproto.instance**

**Explanation**
A request was made for the mechanism by which the route for the specified instance was determined. Values returned will be ‘RIP’ or ‘Other’ depending on whether the route was learned dynamically or was manually entered.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSNMP

**Procedure name**
snmp_lookup

---

**EZB4086I  iprouteage.instance**

**Explanation**
A request was made for the age of the route of the specified instance. The value returned will be the length of time the route has been active.

**System action**
NCPROUTE continues.

**Operator response**
None.
**System programmer response**
None.

**Module**
NRSNMP

**Procedure name**
snmp_lookup

**EZB4087I  iproutemask.instance**

**Explanation**
A request was made for the network or subnetwork mask of route of the specified instance. The value returned will be the network or subnetwork mask value in dotted-decimal notation.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSNMP

**Procedure name**
snmp_lookup

**EZB4088E  An unsupported MIB variable: variable**

**Explanation**
A request was made for an unsupported MIB variable. Only variables in the iproute group are returned. NO_SUCH_NAME will be returned for the unsupported MIB variable.

**System action**
NCPROUTE continues.

**Operator response**
Reissue the request with a valid MIB variable from the iproute group.

**System programmer response**
None.

**Module**
NRSNMP
**Procedure name**

snmp_lookup

**EZB4089E**  An attempt was made to change a MIB variable. SNMP set request is not currently supported.

**Explanation**

An attempt was made to change a MIB variable. Variables in the iproute group may be queried but not changed. An SNMP_GEN_ERR is returned.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

NRSNMP

**Procedure name**

dpi_set

**EZB4090E**  SNMP buffer overrun: number bytes

**Explanation**

An SNMP request will exceed NCPROUTE’s internal buffer size during an edit. This almost always indicates an incorrect request because no supported object identifier is large enough to cause this. The buffer is discarded.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Verify the object identifier being requested is for an iproute MIB variable. If it is, contact IBM software support services.

**Module**

NRSNMP

**Procedure name**

edit_obj

**EZB4125I**  ASN.1 type at count bytes into packet: Class=class Field=field
**Explanation**

An ASN.1 object such as an object identifier or an integer was located in the SNMP packet being displayed. The count is the offset into the packet where the object was located. The object's tag's class bits are formatted and displayed as *class*. An object's class can be viewed as the scope of the object identifier. ASN.1 defines four values:

**Value**

**Explanation**

**Universal**

Well-known tags. NCPROUTE wants this type of tags.

**Application-Wide**

Tags local to the application. NCPROUTE does not have any of these.

**Context-Specific**

Used in ASN.1 constructors. Not applicable to NCPROUTE.

**Private-Use**

NCPROUTE does not use any private tags.

The object's tag's field bits are formatted and displayed as the field. The field is either primitive or constructed.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

NRSNMP

**Procedure name**

decode

**EZB4126E**  Modify command ignored, invalid client address

**Explanation**

The target client specified in the MODIFY command is not valid.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Correct the address for the client target.

**Module**

NRMAIN
**Procedure name**
do_modify

**EZB4127E** Modify command ignored, no session with client *client*

**Explanation**
The target client specified in the MODIFY command is unknown. No session exists between NCPROUTE and the client.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Wait for NCPROUTE to establish a session with the NCP client or respecify the target client using an address known by NCPROUTE.

**Module**
NRMAIN

**Procedure name**
do_modify

**EZB4128I** Reserved for addenda *(type)*

**Explanation**
The type value is not a known object tag type. It was described as reserved for addenda in RFC 1158. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSNMP

**Procedure name**
delete

**EZB4129I** *type*
Explanation
The formatted ASN.1 object tag’s number. NCPROUTE formats Universal tags as defined in RFC 1158. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
decode

EZB4130I Misc type type

Explanation
This object’s class indicates that it is not a well-known type. The unformatted object type is displayed, NCPROUTE only formats objects with a class of Universal.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
decode

EZB4131I Encountered editable OID at offset offset

Explanation
An object identifier was located at the indicated offset into the packet. NCPROUTE will need to translate this OID to another region of the MIB tree so that the agent will recognize the request as being for one of NCPROUTE’s clients and not for the host running the agent.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
decode

EZB4132I  Length = length

Explanation
The length of the decoded ASN.1 object is displayed.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
decode

EZB4133I  Adding type route route destination via gateway gateway, metric metric

Explanation
The indicated route, defined in the client's hlq.ETC.GATEWAYS data set, is added to the NCP's IP routing table. The route to the gateway will not be replaced by a competing RIP route.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
None.

Module
NRSTART

Procedure name
gateways

EJB4134W  Subnetwork mask unknown for destination, using network route route

Explanation
The indicated subnetwork route, defined in the client's GATEWAYS data set, was explicitly coded as a "net" route type. Because the subnetwork mask for the destination subnetwork is unknown, NCPROUTE replaces the subnetwork route with a network route. NCPROUTE currently does not support variable subnetting.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTABLES

Procedure name
nradd

EJB4135E  Invalid host or (sub)network address 'destination'

Explanation
In the line entry for the client's hlq.ETC.GATEWAYS data set, the gateway definition has an incorrect destination address. The destination must be either a resolvable host name or an IP address in dotted notation. The gateway entry is ignored.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the client's hlq.ETC.GATEWAYS data set.
EZB4136W  Second element ignored, changing to ‘active’

Explanation
The second element in the active gateway entry is detected to be incorrect. NCPROUTE will change the element to be ACTIVE since it is likely that the gateway entry is in error. With this change, the active gateway entry is processed normally.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the client's hlq.ETC.GATEWAYS data set.

EZB4138W  Unknown next hop address ipaddr for route destination from router router

Explanation
An unknown next hop address ipaddr was received in a RIP packet for a route to destination from from router. The route is ignored.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Locate the router that produced the packet and correct the problem. It may involve router reconfiguration.

Module
NRSTART

Procedure name
gateways

Module
NRINPUT

Procedure name
rip_input
EZB4139S  tblclr: no host hash table for client client

Explanation
A required host hash table could not be located for the client. NCPROUTE was attempting to remove all routes for the specified client from its tables. Any routes will be left in NCPROUTE’s tables.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRTABLES

Procedure name
tblclr

EZB4141S  tblclr: no network hash table for client client

Explanation
A required network hash table could not be located for the client, but the host hash table has already been found and cleared. NCPROUTE was attempting to remove all network routes for the client from its tables. Any network routes for the client remain in NCPROUTE’s tables.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRTABLES

Procedure name
tblclr

EZB4142E  The transmission of an ‘Ack’ to client client failed.

Explanation
TCPIP detected that an “Acknowledge” PDU could not be delivered successfully before any transmission was performed. The PDU is discarded. A more specific error message will immediately follow.
**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Examine the following error message and follow the documented procedures.

**Module**
NRPDUS

**Procedure name**
send_ack

**EZB4143S**  No SNMP_AGENT statement in profile

**Explanation**
A required SNMP_AGENT statement was missing from the NCPROUTE profile. This statement must identify the host which runs the SNMP agent which NCPROUTE will use to resolve queries. SNMP requests will not be honored by NCPROUTE (or its clients).

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Correct the profile.

**Module**
NRMAIN

**Procedure name**
read_profile

**EZB4144S**  SNMP requests will not be honored.

**Explanation**
One or more severe errors were encountered which will result in a connection with the SNMP agent not being established. Until these errors are resolved, and NCPROUTE is restarted, SNMP requests will not be honored by NCPROUTE and its clients in turn.

**System action**
NCPROUTE continues.
Operator response
None.

System programmer response
Correct any previously identified errors.

Module
NRMAIN

Procedure name
read_profile

EZB4145S  * No SNMP_COMMUNITY statement in NCPROUTE profile

Explanation
A required SNMP_COMMUNITY statement is missing from the NCPROUTE profile. This statement identifies the SNMP community which will be used when forwarding SNMP requests to the agent. SNMP requests will not be honored.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the profile and restart NCPROUTE.

Module
NRMAIN

Procedure name
read_profile

EZB4146W  Required NCPROUTE profile dataset (profile) not found

Explanation
The optional profile configuration data set for NCPROUTE could not be opened successfully. This message indicates which data set the open was attempted on.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
Verify that the profile data set is defined in //DD:NCPRPROF of the NCPROUTE start proc JCL and is accessible by NCPROUTE. If the data set is sequential, ensure that the FREE=CLOSE parameter is specified. Do not specify this parameter if the data set is partitioned.

Module
NRMAIN

Procedure name
read_profile

EZB4147S  Unknown keyword (keyword) in PROFILE

Explanation
An unknown keyword *keyword* was used in a statement in the NCPROUTE profile. The statement is ignored.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the NCPROUTE profile.

Module
NRMAIN

Procedure name
read_profile

EZB4148E  client *client*: failed attempt to add route to *destination*.

Explanation
NCPROUTE’s attempt to add a route to the client was rejected by the client. Most likely the client’s route tables are full so that it could not accept any more routes, or the NCP is attempting to add a route specified as PERM during generation.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Verify the routing table size values in the NUMROUTE keyword in the IPOWER statement of the NCP generation. Increase the values, if necessary.
Module
NRPDUS

Procedure name
add_fail

EZB4149E  client client: failed attempt to delete route to destination.

Explanation
Client client rejected NCPROUTE’s attempt to delete a route to destination from the client’s tables. Most likely the route did not exist in the client’s tables.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRPDUS

Procedure name
delete_fail

EZB4150I  End of GATEWAYS processing

Explanation
Processing is completed for the GATEWAYS data set member.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSTART

Procedure name
gateways

EZB4151T  Out of memory while processing GATEWAYS
**Explanation**
NCPROUTE has exhausted available storage while processing a client's GATEWAYS data set member.

**System action**
NCPROUTE ends abnormally.

**Operator response**
None.

**System programmer response**
Increase NCPROUTE's region size and restart. Take into consideration that storage requirements are based on the number of clients being served and the number of routes being managed. If the problem still cannot be resolved, contact IBM software support services.

**Module**
NRSTART

**Procedure name**
gateways

**EZB4152I** Adding active gateway ip_addr, metric metric

**Explanation**
The indicated active gateway, which is defined in the client's hlq.ETC.GATEWAYS data set, is added to NCPROUTE's routing table. The route to the active gateway will be treated as a network interface.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSTART

**Procedure name**
gateways

**EZB4154E** Invalid option: option

**Explanation**
In the line entry for the client's hlq.ETC.GATEWAYS data set, the NCPROUTE's server options definition has an incorrect option. Although other options may be processed normally, the incorrect option is ignored.
System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the client's hlq.ETC.GATEWAYS data set. See z/OS Communications Server: IP Configuration Reference for more information.

Module
NRSTART

Procedure name
gateways

EZB4155E Invalid default router value: value

Explanation
In the line entry for the client's hlq.ETC.GATEWAYS data set, the NCPROUTE's server options definition has an incorrect default router value. The incorrect default router value is ignored and the default value is used.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the client's hlq.ETC.GATEWAYS data set by specifying a valid default router value. See z/OS Communications Server: IP Configuration Reference for more information.

Module
NRSTART

Procedure name
gateways

EZB4156E Invalid trace level: level

Explanation
In the line entry for the client's hlq.ETC.GATEWAYS data set, the NCPROUTE's server options definition has an incorrect trace level. The incorrect trace level is ignored and the default value is used.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
Correct the client's hlq.ETC.GATEWAYS data set by specifying a valid trace level value. See z/OS Communications Server: IP Configuration Reference for more information.

Module
NRSTART

Procedure name
gateways

EZB4157E Invalid type value: value

Explanation
In the line entry for a client's hlq.ETC.GATEWAYS data set, the options definition contains an invalid value. Although other options may be processed normally, the invalid option is ignored.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
Correct the client's hlq.ETC.GATEWAYS data set. See z/OS Communications Server: IP Configuration Reference for more information.

Module
NRSTART

Procedure name
gateways

EZB4158I Global tracing at all levels suppressed

Explanation
Trace levels for all NCP clients are not displayed. This includes the trace levels specified in the clients' hlq.ETC.GATEWAYS data set.

System action
NCPROUTE continues.

Operator response
None.
System programmer response
None.

Module
NRTRACE

Procedure name
global_trace_start

EZB4159I  Global tracing actions started

Explanation
Trace levels for all NCP clients have been advanced to the "actions" level, which causes messages to be issued for actions such as adding, changing, or deleting a route. Additional messages for actions such as waiting for incoming packets and dynamic updates are also issued. The trace levels specified in the client's hlq.ETC.GATEWAYS data sets are also advanced to this level.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRTRACE

Procedure name
global_trace_start

EZB4160I  Global tracing packets started

Explanation
Trace levels for all NCP clients have been advanced to the "packets" level, which displays the types of packets sent and received in addition to the output displayed at the "actions" level. The trace levels specified in the client's hlq.ETC.GATEWAYS data sets are also advanced to this level.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
Module
NRTRACE

Procedure name
global_trace_start

EZB4161T  Global trace levels exceeded maximum of 2 -t's. For each client, use the MODIFY command or GATEWAYS dataset to specify additional tracing options.

Explanation
An incorrect number of trace levels (-t's) was passed from SEZAINST(NCPROUTE) start proc JCL. The parameter(s) could either be passed from the command line parameters or from the default parameter list in the start proc JCL.

System action
NCPROUTE exits.

Operator response
None.

System programmer response
Specify a correct number of -t's in the command line parameters or in the default parameter list of the start proc JCL. If a higher trace level is required, specify the trace level in the options statement of a client's hlq,ETC.GATEWAYS data set. Another option is to use the MODIFY command to increase the trace levels for the client after NCPROUTE has started and established a session with the client. See z/OS Communications Server: IP Configuration Reference for more information.

Module
NRMAIN

Procedure name
parse_parms

EZB4162I  Deferring add route to destination

Explanation
The addition of a new route to the specified destination in the NCP client's routing table is deferred until after the NCP client has finished initialization. The new route is copied to a static buffer, which is used after the Inactive Interface List PDU has been received from the NCP client. NCPROUTE will perform the add from the static buffer.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
Module
NRPDUS

Procedure name
defer_add

**EZB4163E** Too many deferred routes

**Explanation**
The number of static buffers to hold the new routes for deferred addition to the NCP clients' routing tables has exceeded the maximum of 500. This may happen when multiple NCP clients have not completed their initialization processes at the same time. At this time, the new routes are not added to the NCP client's routing tables. The static buffers do not become available until after NCPROUTE has received the Inactive Interface List PDUs from the NCP clients. These PDUs tells NCPROUTE that the NCP clients have completed their initialization processes.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Allow sufficient time for the NCP clients to complete their initialization processes so that the new routes can be added to the client's routing tables. Contact IBM software support services if the problem persists.

Module
NRPDUS

Procedure name
defer_add

**EZB4164I** Init: compare with client

**Explanation**
The client's IP address is being compared with one of NCPROUTE's session table entries for an exact match.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.
**Procedure name**
ncp_initialized

**EZB4165E  Client client not in session table**

**Explanation**
The client's IP address was not found in the session table. This indicates that NCPROUTE has not established a session with the NCP client. NCPROUTE will not manage the routes for the client until a "Hello" PDU has been received for session establishment.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Allow sufficient time for the NCP client to issue the "Hello" PDUs so that a session can be established with NCPROUTE. If a Hello PDU was received and the problem persists, contact IBM software support services.

**Module**
NRPDUS

**Procedure name**
ncp_initialized

**EZB4166I  Session with client client started**

**Explanation**
NCPROUTE has successfully established a session with the specified NCP client through the handshaking process and is waiting to receive an Inactive Interface List PDU from the client. This PDU indicates that the client has completed initialization and is ready for route table management by NCPROUTE.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRPDUS

**Procedure name**
recv_hello

**EZB4167E  SNMP requester requester is not reachable by NCP client client**
**Explanation**

NCPROUTE was ready to transport the SNMP response packet to the NCP client to be forwarded to the SNMP requester, but the route to the SNMP requester was detected to be unavailable. A possible cause is that the NCP client's interface to the SNMP requester became inactive. The SNMP response packet is discarded.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Verify the status of the NCP client's interface to the SNMP requester. If the interface was inactive, activate the interface so that a route can be created; otherwise, determine whether the route to the SNMP requester was timed out. If the problem cannot be corrected, contact IBM software support services.

**Module**

NRSNMP

**Procedure name**

snmp_response

---

**EZB4168I  SNMP lookup failed: reason**

**Explanation**

NCPROUTE could not obtain the routing information based upon the SNMP request packet sent by the SNMP requester. In this case, NO_SUCH_NAME is returned to the SNMP requester. Reasons for the failure are:

**Reason**

**Explanation**

NO INSTANCES PROVIDED
The target route was not specified in the SNMP request; however, this is an exception to SNMP GET_NEXT requests.

ROUTE TABLE EMPTY
There were no routing table entries for this interface based upon the target route.

ROUTE NOT FOUND
The target route was not found in the routing table.

**System action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

Verify the accuracy of the target route in the SNMP request. If the problem cannot be corrected, contact IBM software support services.

**Module**

NRSNMP
**procedure name**

snmp_lookup

**EZB4172I  SNMP reply sent to NCP client client**

**Explanation**

NCPROUTE is sending the reply containing the SNMP response packet to the NCP client to be forwarded to the SNMP requester.

**system action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

NRSNMP

**procedure name**

snmp_response

**EZB4173I  SNMP get_next: no session with client client**

**Explanation**

During processing of the SNMP GET_NEXT request, NCPROUTE has detected that there was no session with the NCP client. The SNMP request packet is discarded.

**system action**

NCPROUTE continues.

**Operator response**

None.

**System programmer response**

If the session with the NCP client appears to have terminated, examine previous messages to determine the cause. When necessary, take corrective actions to allow NCPROUTE to process the SNMP requests for the NCP client.

**Module**

NRSNMP

**procedure name**

get_next

**EZB4174I  SNMP get_next: searching for next route after target**
Explanation
During processing of the SNMP GET_NEXT request, NCPROUTE is searching for the next route after the target route in the routing tables for the NCP client.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
get_next

EJB4175I SNMP get_next: found next route ip_addr

Explanation
During processing of the SNMP GET_NEXT request, NCPROUTE has found the next route entry after the target route in the routing tables for the NCP client. The entry contains the next route's ip_addr.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
get_next

EJB4176I SNMP get_next: compare with target

Explanation
The target route specified in the SNMP GET_NEXT request packet received by NCPROUTE from an NCP client is being compared with one of NCPROUTE's routing table entries for an exact match.

System action
NCPROUTE continues.
Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
get_next

EZB4177I SNMP get_next: addresses matched

Explanation
A match was found for the target route specified in the SNMP GET_NEXT request packet with one of NCPROUTE’s routing table entries for an NCP client.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRSNMP

Procedure name
g et_next

EZB4178I RIP2 authentication action at level level (interface)

Explanation
RIP Version 2 authentication is enabled or disabled at the specified level for an interface for all interfaces in the NCP client, or for all NCP clients.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.
Module
NRMAIN, NRSTART

Procedure name
read_profile, ParseOptions

EZB4179E    SNMP object length exceeded maximum 64K

Explanation
During the object data conversion process for an SNMP packet, NCPROUTE detected an incorrect length value in
the header portion of the object data. NCPROUTE could not continue processing for the value has exceeded the
maximum length of 64K. The SNMP packet may have been built incorrectly.

System action
NCPROUTE exits.

Operator response
None.

System programmer response
Contact IBM software support services.

Module
NRSNMP

Procedure name
store_int

EZB4180I    Packet from router router ignored (filtered out)

Explanation
NCPROUTE is ignoring the RIP packet as a result of being filtered out according to a RIP input or output filter
defined in the NCP clients's gateways data set.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
None.

Module
NRINPUT

Procedure name
rip_input
EZB4181I  Interface *interface* skipped, interface is multicast-incapable

**Explanation**
The interface *interface* is skipped because the interface is not capable of multicasting RIP Version 2 packets. The RIP2 supply control has been configured for the interface and disallows the use of broadcasting RIP packets over the interface.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
When permitted, use the RIP2M supply control option to allow broadcasting RIP Version 1 packets over multicast-incapable interfaces and multicasting Version 2 packets over the multicast-capable interfaces.

**Module**
NROUTPUT

**Procedure name**
toall_ifs

EZB4182I  SNMP request received from NCP client *client*

**Explanation**
An SNMP query request was received from an NCP client for processing. The NCP client had received the query request from its SNMP client or the SNMP requester.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSNMP

**Procedure name**
snmp_input

EZB4191I  SNMP get-next: next route not found (*reason*)
Explanation
NCPROUTE is processing the SNMP GET_NEXT request but it could not obtain the routing information for the next route. In this case, NO_SUCH_NAME is returned to the SNMP requester. Reasons for the unavailable route are:

<table>
<thead>
<tr>
<th>Reason</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOT IN TABLE</td>
<td>The target route was not found in the routing table. This implies that the route entry for the next route cannot be determined.</td>
</tr>
</tbody>
</table>

END OF TABLE
The target route was found at the end of the routing table but there were no more route entries for the next route.

System action
NCPROUTE continues.

Operator response
None.

System programmer response
These reasons require the following responses:

<table>
<thead>
<tr>
<th>Reason</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOT IN TABLE</td>
<td>Verify the accuracy of the target route in the SNMP request.</td>
</tr>
</tbody>
</table>

END OF TABLE
None.

If the problem cannot be corrected, contact IBM software support services.

Module
NRSNMP

Procedure name
snmp_lookup

EZB4194I    SNMP sub-agent received DPI request

Explanation
NCPROUTE is performing the role as an SNMP sub-agent and is in session with the SNMP agent over the Distributed Program Interface (DPI). NCPROUTE has received a DPI request from the SNMP agent for processing.

System action
NCPROUTE continues.

Operator response
None.
**System programmer response**
None.

**Module**
NRSNMP

**Procedure name**
dpi_in

**EZB4195I  Option(s): options**

**Explanation**
Additional NCPROUTE options, specified in a client’s hlq.ETC.GATEWAYS data set member, are being processed. Some options may be overridden by the parameter list in the SEZAINST(NCPROUT) start proc JCL. See z/OS Communications Server: IP Configuration Reference for more information.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSTART

**Procedure name**
gateways

**EZB4196I  Opening NCPROUTE profile data set datasetname**

**Explanation**
The specified NCPROUTE profile data set is being opened. Entries in the data set are read in for input.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.
**Procedure name**
read_profile

**EZB4197E**  **Invalid route type 'type'**

**Explanation**
In the line entry for the NCP client's hlq.ETC.GATEWAYS data set, the gateway definition has an incorrect route type. Allowable route types are HOST for host route, NET for network or subnetwork route, and ACITVE for a route to be treated as a network interface. The gateway entry is ignored.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
Correct the NCP client's hlq.ETC.GATEWAYS data set.

**Module**
NRSTART

**Procedure name**
gateways

**EZB4198I**  **(no etc.gateway definitions)**

**Explanation**
The NCP client's hlq.ETC.GATEWAYS data set contains no optional gateway definitions.

**System action**
NCPROUTE continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
NRSTART

**Procedure name**
gateways
Chapter 7. EZB6xxxx messages

EZB6473I  TCP/IP STACK FUNCTIONS INITIALIZATION COMPLETE.

Explanation
TCP/IP has been successfully initialized.

System action
TCP/IP continues.

Operator response
None.

System programmer response
None.

Module
EZBTTINI
Chapter 8. EZB8xxxx messages

Messages deleted in V2R2:
• EZB8801I

**EZB8801I  probeinm ATTEMPTED, FFST NOT AVAILABLE.**

**Explanation**
TCP/IP encountered an anomaly and attempted to execute a FFST probe. FFST was not available to service the request. The request was handled by furnishing the identification of the probe.

In the message text:

*probeinm*
  The probe identifier.

**System action**
TCP/IP continues.

**Operator response**
Determine why FFST is not available. Gather what probe documentation is available and contact IBM software support services.

**System programmer response**
Determine why FFST is not available. Gather what probe documentation is available and contact IBM software support services.

**Module**
EZBITZPF

**Automation**

This message was deleted in z/OS V2R2.
Chapter 9. EZB9xxxx messages

Common messages

This section contains messages that are called by several application and function components. When these messages are called, only the message text will be appended to the message that called it. The message will inherit the prefix (EZA or EZB) of the message that called it. They are documented fully in z/OS Communications Server: IP Messages Volume 1 (EZA).

The following is a complete list of these common messages.

- EZB9395I
- EZB9396I
- EZB9397I
- EZB9398I
- EZB9399I
- EZB9400I
- EZB9401I
- EZB9402I
- EZB9403I
- EZB9404I
- EZB9405I
- EZB9406I
- EZB9407I
- EZB9408I
- EZB9409I
- EZB9410I
- EZB9411I
- EZB9412I
- EZB9413I
- EZB9414I
- EZB9415I
- EZB9416I
- EZB9417I
- EZB9418I
- EZB9419I
- EZB9420I
- EZB9421I
- EZB9422I
- EZB9423I
- EZB9424I
- EZB9428I
Chapter 10. EZBHxxxx messages

EZBH001I  TCP/IP Event Trace (SYSTCPIP) is active with default options

Explanation
Check CSTCP_SYSTCPIP_CTRACE_tcpipstackname ran successfully and found no exceptions. The check determined that TCP/IP event trace (SYSTCPIP) is active on the stack with the default options. The check name includes the job name of the TCP/IP stack as a suffix.

System action
The system continues processing.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBHCCK1

Example
None.

EZBH002E  TCP/IP Event Trace (SYSTCPIP) is active with non-default options

Explanation
Check CSTCP_SYSTCPIP_CTRACE_tcpipstackname determined that TCP/IP event trace (SYSTCPIP) is active on the stack with options other than the default options, which can result in performance degradation. The check name includes the job name of the TCP/IP stack as a suffix.

System action
The system continues processing. However, eventual action might need to be taken to prevent performance degradation.
Operator response
Contact the system programmer.

System programmer response
Issue the DISPLAY TRACE,COMP=SYSTCPIP,SUB=(jobname) command to display the active SYSTCPIP trace options. Deactivate any options that no longer need to be active. If problem documentation is not being gathered, only the default options (MINIMUM, INIT, OPCMDS, or OPMGS) should be active. See the information about displaying component trace status for information about displaying active trace options and the information about specifying trace options for information about modifying trace options in z/OS Communications Server: IP Diagnosis Guide.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBHCCK1

Example
None.

EZBH003E   TCPMAXRCVBUFRSIZE bufsize is less than the deftype specified minval

Explanation
Check CSTCP_TCPMAXRCVBUFRSIZE_tcpipstackname determined that the configured TCP maximum receive buffer size is less than the minimum specified for the check.

If owner is specified in the message text, the configured value is not sufficient to provide optimal support to the z/OS Communications Server FTP Server.

The check name includes the jobname of the TCP/IP stack as a suffix.

In the message text:

bufsize
The value specified for the TCPMAXRCVBUFRSIZE parameter on the TCPCONFIG statement in the TCP/IP profile or modified using the VARY TCPIP,OBEYFILE command.

deftype
Possible values are:

installation
The check parameters for this check have not been overridden.

owner
The check parameters for this check have been overridden.

minval
The check parameter value against which the TCPMAXRCVBUFRSIZE value is compared.
**System action**
The system continues processing. However, eventual action might need to be taken to provide optimal support to the z/OS Communications Server FTP server.

**Operator response**
Contact the system programmer.

**System programmer response**
Optimally, the z/OS Communications Server FTP server needs a buffer size of 180 KB for data connections. If z/OS Communications Server FTP server is being used on the stack reporting the problem, modify the TCPMAXRCVBUFSIZE parameter on the TCPCONFIG statement in your TCP/IP profile to specify a value of at least 180 KB. To make this change effective immediately, use the VARY TCPIP,,OBEYFILE command, specifying a profile containing the modified TCPCONFIG statement.

See the information about the TCPCONFIG statement in z/OS Communications Server: IP Configuration Reference for more information about the TCPMAXRCVBUFSIZE parameter.

See the information about the VARY TCPIP,,OBEYFILE command in z/OS Communications Server: IP System Administrator's Commands for more information about the VARY TCPIP,,OBEYFILE command.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBHCCK1

**Example**
EZH003E TCPMAXRCVBUFSIZE 120K is less than the owner specified 180K

EZH004I TCPMAXRCVBUFSIZE bufsize satisfies the deftype limit of minval

**Explanation**
Check CSTCP_TCPMAXRCVBUFSIZE_{tcpipstackname} ran successfully and found no exceptions. The check determined that the value specified for the TCPMAXRCVBUFSIZE parameter on the TCPCONFIG statement in the TCP/IP profile is in the limit specified for this check.

If owner is specified in the message text, the configured value is sufficient to provide optimal support to the z/OS Communications Server FTP Server.

The check name includes the job name of the TCP/IP stack as a suffix.

In the message text:

**bufsize**
The value specified for the TCPMAXRCVBUFSIZE parameter on the TCPCONFIG statement in the TCP/IP profile or modified using the VARY TCPIP,,OBEYFILE command.

**deftype**
Possible values are:
The check parameters for this check have not been overridden.

The check parameters for this check have been overridden.

The check parameter value against which the TCPMAXRCVBUFRSIZE value is compared.

The specified check completes. System processing continues.

None.

None.

Not applicable.

Not applicable.

z/OS Communications Server TCP/IP

EZBHCCK1

GLOBALCONFIG SYSPLEXMONITOR RECOVERY is specified when IPCONFIG DYNAMICXCF or IPCONFIG6 DYNAMICXCF is configured

Check CSTCP_SYSPLEXMON_RECOV_tcpipstackname ran successfully and found no exceptions. The check determined that the SYSPLEXMONITOR RECOVERY parameter was specified when the DYNAMICXCF option was configured for this stack.

The check name includes the job name of the TCP/IP stack as a suffix.

The system continues processing.

Not applicable.

Not applicable.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBHCCK1

Routing code
Not applicable.

Descriptor code
Not applicable.

Example
Not applicable.

GLOBALCONFIG SYSPLEXMONITOR RECOVERY was not specified when IPCONFIG DYNAMICXCF or IPCONFIG6 DYNAMICXCF was configured

Explanation
Check CSTCP_SYSPLEXMON_RECOV_tcpipstackname determined that the RECOVERY option was not specified for the GLOBALCONFIG SYSPLEXMONITOR parameter when IPCONFIG DYNAMICXCF or IPCONFIG6 DYNAMICXCF was specified in the TCP/IP profile.

IBM suggests that the SYSPLEXMONITOR RECOVERY option be specified when DYNAMICXCF is specified in the TCP/IP profile. Specifying this option allows a TCP/IP stack in a sysplex to perform internal checks and, if it is not healthy, remove itself from the sysplex, allowing a healthy backup TCP/IP stack to take over the ownership of the DVIPA interfaces to enable continued availability to applications.

The check name includes the job name of the TCP/IP stack as a suffix.

System action
The system continues processing.

Operator response
Contact the system programmer.

System programmer response
Change the GLOBALCONFIG SYSPLEXMONITOR parameter to specify RECOVERY when your TCP/IP profile specifies IPCONFIG DYNAMICXCF or IPCONFIG6 DYNAMICXCF. In z/OS Communications Server: IP Configuration Reference see the following information:

• The GLOBALCONFIG configuration statement for more information about the SYSPLEXMONITOR RECOVERY parameter.
• The IPCONFIG and IPCONFIG6 configuration statements for more information about the DYNAMICXCF parameter.
User response
Not applicable.

Problem determination
Use the NETSTAT CONFIG/-f command to display the current configuration setting for DYNAMICXCF and SYSPLEXMONITOR RECOVERY.

Source
z/OS Communications Server TCP/IP

Module
EZBHCCK1

Routing code
Not applicable.

Descriptor code
12

Example
Not applicable.

EZBH007I The port range defined for CINET use has been reserved for OMVS on this stack.

Explanation
Check CSTCP_CINET_PORTRNG_RSV_tcpipstackname ran successfully and found no exceptions. It determined that the range of ports defined for CINET use in the BPXPRMxx parmlib member has been reserved on this stack for OMVS, using the PORTRANGE TCP/IP profile statement.

The check name includes the jobname of the TCP/IP stack as a suffix.

System action
The system continues processing.

Operator response
Not applicable.

System programmer response
Not applicable.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP
Module
EZBHCK1

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
Not applicable.

EZBH008E The port range defined for CINET use has not been reserved for OMVS on this stack.

Explanation
Check CSTCP_CINET_PORTRNG_RSV_{tcpipstackname} determined that the port range defined for CINET use in the BPXPRMxx parmlib member is not reserved for OMVS on this stack.

IBM suggests that the port range specified by the INADDRANYPORT and INADDRANYCOUNT parameters in the BPXPRMxx parmlib member for use by CINET should be reserved for OMVS using the PORTRANGE TCP/IP profile statement. This prevents a TCP/IP stack from allocating a port that CINET might subsequently attempt to use, which might result in an ABEND and the message:

BPXF219I A SOCKETS PORT ASSIGNMENT CONFLICT EXISTS BETWEEN UNIX SYSTEM SERVICES AND {tcpipstackname}

See the section on ephemeral port reservations for multiple instances of TCP/IP in z/OS Communications Server: IP Configuration Guide for more information about the reservations needed.

See the information about the PORTRANGE statement in z/OS Communications Server: IP Configuration Reference for more information about the PORTRANGE statement.

See for z/OS MVS Initialization and Tuning Reference more information on the INADDRANYPORT and INADDRANYCOUNT parameters.

The check name includes the jobname of the TCP/IP stack as a suffix.

System action
The system continues processing.

Operator response
Determine the CINET INADDRANY port range by issuing

D OMVS,CINET

Contact the responsible system programmer to create a data set containing the PORTRANGE statements needed. Once available issue:

V TCPIP,{tcpipstackname},OBEYFILE,filename

In the message text:

tcpipstackname The name of the TCP/IP stack.
filename

The name of the OBEY file to reserve the CINET INADDRANY port range.

See the information about the VARY TCPIP,OBEYFILE command in z/OS Communications Server: IP System Administrator's Commands for more information about the VARY TCPIP,OBEYFILE command.

System programmer response

Determine the CINET INADDRANY port range by issuing:

```
D OMVS,CINET
```

Change the TCP/IP profile to include a PORTRANGE statement for both TCP and UDP to reserve the port range defined by the INADDRANYPORT and INADDRANYCOUNT parameters in the BPXPRMxx parmlib member for use by OMVS. Also, place these statements in a separate data set for use in an OBEYFILE command to have these reservations take effect immediately. For more information on OBEYFILE command, see the operator response section of this message.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP

Module

EZBHCCK1

Routing code

Not applicable.

Descriptor code

3

Automation

Not applicable.

Example

Not applicable.

EZH009I This check is not applicable in the current environment. This check is applicable only in a CINET environment.

Explanation

Check CSTCP_CINET_PORTRNG_RSV_tcpipstackname is not applicable in the current environment. It is applicable only in a CINET environment, and determines whether the port range defined for CINET use by the INADDRANYPORT and INADDRANYCOUNT parameters in the BPXPRMxx parmlib member are reserved for OMVS by specifying a PORTRANGE statement in the TCP/IP profile for this stack.

The check name includes the jobname of the TCP/IP stack as a suffix.
The current number of IPv4 indirect routes is below or equal to the maximum threshold
(Current = current_val, High = high_val, Maximum = maximum_val)

Explanation
The CSTCP_IPMAXRT4_TCPIPstackname check ran successfully and found no exceptions. The check determined that the current number of IPv4 indirect static and dynamic routes in the TCP/IP stack routing table is within the maximum threshold value for this check. This message is issued after TCP/IP initialization and at the specified time intervals using the INTERVAL parameter of the IBM Health Checker for z/OS.

The check name includes the job name of the TCP/IP stack as a suffix.

current_val is the current number of IPv4 indirect routes in the TCP/IP stack routing table.

high_val is the highest recorded number of IPv4 indirect routes in the TCP/IP stack routing table during the time interval as defined in the INTERVAL parameter for the IBM Health Checker for z/OS.

maximum_val is the configured or default maximum threshold for the number of IPv4 indirect routes in the TCP/IP stack routing table as defined in the IPMAXRT4 parameter for the IBM Health Checker for z/OS.
System action
The system continues processing.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBHCK1

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

EZBH010I The current number of IPv4 indirect routes is below or equal to the maximum threshold
(CURRENT = 1000, HIGH = 1500, MAXIMUM = 2000)

EZBH011E The current number of IPv4 indirect routes exceeds the maximum threshold (CURRENT =
current_val, HIGH = high_val, MAXIMUM = maximum_val)

Explanation
The CSTCP_IPMAXRT4_TCPIP_STACKNAME check determined that the current number of IPv4 indirect static and
dynamic routes in the TCP/IP stack routing table is greater than the maximum threshold value for this check. The high number of routes might cause high CPU consumption from routing changes and less than optimal operations in OMPROUTE and the TCP/IP stack. Actions should be taken to reduce the number of routes.

The check name includes the job name of the TCP/IP stack as a suffix.

current_val is the current number of IPv4 indirect routes in the TCP/IP stack routing table.

high_val is the highest recorded number of IPv4 indirect routes in the TCP/IP stack routing table during the time interval as defined in the INTERVAL parameter for the IBM Health Checker for z/OS.
maximum_val is the configured or default maximum threshold for the number of IPv4 indirect routes in the TCP/IP stack routing table as defined in the IPMAXRT4 parameter for the IBM Health Checker for z/OS.

**System action**
The system continues processing.

**Operator response**
Contact the system programmer.

**System programmer response**
To reduce the number of indirect OSPF routes, IBM suggests that the OSPF areas that contain the z/OS Communications Server application host or sysplex be configured as stub areas for optimal performance. Stub areas are used so that IPv4 route summaries from other areas are not flooded into the stub areas by the area border routers. See the minimizing the routing responsibility of z/OS Communications Server information in z/OS Communications Server: IP Configuration Guide.

Route filtering options for the address ranges in the areas are also available to control the OSPF route advertisements. See the AREA statement and the RANGE statement in z/OS Communications Server: IP Configuration Reference for more information about the Stub_Area and Advertise parameters.

To reduce the number of indirect RIP or static routes, IBM suggests the configuration of RIP or static networks that contain the z/OS Communications Server application host or sysplex use route summarization and/or filters for optimal performance. Use of network-specific or default routes that represent the more specific indirect routes provides the route summarization.

Use of the RIPv2 option provides network masks in the RIP advertisements. Filtering options are also available to control the RIP route advertisements. See the RIP_INTERFACE statement in z/OS Communications Server: IP Configuration Reference for information on RIPv2 and the receive filter parameters.

See the BEGINROUTES statement in z/OS Communications Server: IP Configuration Reference for information on defining IPv6 prefixes or default routes. Modify the routes as necessary for route summarization. Then issue a VARY TCPIP,,OBEYFILE command with an OBEY file that contains the modified BEGINROUTES block.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
EZBHCK1

**Routing code**
Not applicable.

**Descriptor code**
3
Automation
Not applicable.

Example

EZH011E The current number of IPv4 indirect routes exceeds the maximum threshold (Current = 2500, High = 3000, Maximum = 2000)

EZH012I The current number of IPv6 indirect routes is below or equal to the maximum threshold (Current = current_val, High = high_val, Maximum = maximum_val)

Explanation
The CSTCP_IPMAXRT6_TCPIP_stackname check ran successfully and found no exceptions. The check determined that the current number of IPv6 indirect static and dynamic routes in the TCP/IP stack routing table is within the maximum threshold value for this check. This message is issued after TCP/IP initialization and at the specified time intervals using the IBM Health Checker's INTERVAL parameter.

The check name includes the job name of the TCP/IP stack as a suffix.

current_val is the current number of IPv6 indirect routes in the TCP/IP stack routing table.

high_val is the highest recorded number of IPv6 indirect routes in the TCP/IP stack routing table during the time interval as defined in the INTERVAL parameter for the IBM Health Checker for z/OS.

maximum_val is the configured or default maximum threshold for the number of IPv6 indirect routes in the TCP/IP stack routing table as defined in the IPMAXRT6 parameter for the IBM Health Checker for z/OS.

System action
The system continues processing.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBHCCK1

Routing code
Not applicable.
Descriptor code
Not applicable.

Automation
Not applicable.

Example

EZH012I The current number of IPv6 indirect routes is below or equal to the maximum threshold (Current = 500, High = 750, Maximum = 2000)

EZH013E The current number of IPv6 indirect routes exceeds the maximum threshold (Current = current_val, High = high_val, Maximum = maximum_val)

Explanation
The CSTCP_IPMAXRT6_TCPIPstackname check determined that the current number of IPv6 indirect static and dynamic routes in the TCP/IP stack routing table is greater than the maximum threshold value for this check. The high number of routes might cause high CPU consumption from routing changes and less than optimal operations in OMPROUTE and the TCP/IP stack. Actions should be taken to reduce the number of routes.

The check name includes the job name of the TCP/IP stack as a suffix.

current_val is the current number of IPv6 indirect routes in the TCP/IP stack routing table.

high_val is the highest recorded number of IPv6 indirect routes in the TCP/IP stack routing table during the time interval as defined in the INTERVAL parameter for the IBM Health Checker for z/OS.

maximum_val is the configured or default maximum threshold for the number of IPv6 indirect routes in the TCP/IP stack routing table as defined in the IPMAXRT6 parameter for the IBM Health Checker for z/OS.

System action
The system continues processing.

Operator response
Contact the system programmer.

System programmer response
To reduce the number of OSPF indirect routes, IBM suggests that the OSPF areas that contain the z/OS Communications Server application host or sysplex be configured as stub areas for optimal performance. Stub areas are used so that IPv6 prefixes from other areas are not flooded into the stub areas by the area border routers. See the minimizing the routing responsibility of z/OS Communications Server information in z/OS Communications Server: IP Configuration Guide.

Route filtering options for the address ranges in the areas are also available to control the OSPF route advertisements. See the IPv6_AREA statement and the IPv6_RANGE statement in z/OS Communications Server: IP Configuration Guide for more information about the Stub_Area and Advertise parameters.

To reduce the number of indirect RIP or static routes, IBM suggests the configuration of RIP or static networks that contain the z/OS Communications Server application host or sysplex use route summarization and/or filters for optimal performance. Use of IPv6 prefixes or default routes that represent the more specific indirect routes provides the route summarization.

Filtering options are also available to control the RIP route advertisements. See the IPv6_RIP_INTERFACE statement in z/OS Communications Server: IP Configuration Guide for information on the receive filter parameters.
See the `BEGINROUTES` statement in *z/OS Communications Server: IP Configuration Reference* for information on defining IPv6 prefixes or default routes. Modify the routes as necessary for route summarization. Then issue a `VARY TCPIP,OBEYFILE` command with an OBEY file that contains the modified `BEGINROUTES` block.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBHCCK1

**Routing code**
Not applicable.

**Descriptor code**
3

**Automation**
Not applicable.

**Example**

```
EZBH013E The current number of IPv6 indirect routes exceeds the maximum threshold
(Current = 4000, High = 3000, Maximum = 2000)

EZBH014I GLOBALTCPIPDATA is specified when AUTOQUIESCE is specified.
```

**Explanation**
Check CSRES_AUTOQ_GLOBALTCPIPDATA ran successfully and found no exceptions. The check detected that the GLOBALTCPIPDATA resolver setup statement is specified when the AUTOQUIESCE operand is specified on the UNRESPONSIVETHRESHOLD resolver setup statement.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

---

502 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBHCCCK2

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

EZBH014I - GLOBALTCPIPDATA is specified when AUTOQUIESCE is specified.

EZBH015E GLOBALTCPIPDATA was not specified when AUTOQUIESCE was specified.

Explanation
Check CSRES_AUTOQ_GLOBALTCPIPDATA detected that the GLOBALTCPIPDATA resolver setup statement was not specified when the AUTOQUIESCE operand was specified on the UNRESPONSIVETHRESHOLD resolver setup statement. The AUTOQUIESCE operand was ignored.

System action
The resolver uses the UNRESPONSIVETHRESHOLD percentage value to perform the network operator notification function instead of performing the autonomic quiescing of unresponsive name servers function.

Operator response
Contact the system programmer.

System programmer response
- If you do not want unresponsive name servers to be automatically quiesced, perform one of the following actions:
  - Remove the AUTOQUIESCE operand from the UNRESPONSIVETHRESHOLD statement, but leave the threshold percentage coded on the statement.
  - Remove the UNRESPONSIVETHRESHOLD statement completely. The network operator notification function will run by default.
  - Leave the resolver setup file unchanged. You will continue to see message EZBH015E every time the resolver is started or a MODIFY RESOLVER,REFRESH command is issued, but the autonomic quiescing function will not be active.
- If you want unresponsive name servers to be automatically quiesced, perform the following actions:
If you do not already have a global TCPIP.DATA file, create one. Code the appropriate resolver-related TCPIP.DATA statements in the global TCPIP.DATA file you just created.

Code the GLOBALTCPIPDATA statement in the resolver setup file, specifying the name of the global TCPIP.DATA file to be used.

If you have corrected the resolver setup file, instruct the operator to issue the MODIFY RESOLVER,REFRESH,SETUP=setup_file_name command to activate the changes.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBHCCCK2

**Routing code**
Not applicable.

**Descriptor code**
3

**Automation**
This message is a candidate for automation in cases where different resolver setup files are used to enable different resolver functions.

**Example**

```
EZBH015E - GLOBALTCPIPDATA was not specified when AUTOQUIESCE was specified
```

**Explanation**
Check CSRES_AUTOQ_GLOBALTCPIPDATA is not applicable when the AUTOQUIESCE operand is not specified on the UNRESPONSIVETHRESHOLD resolver setup statement.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
None.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZHHCCK2

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
This message is not a candidate for automation.

Example
EZBH016I - This check is not applicable when AUTOQUIESCE is not specified

EZBH017I The resolver timeout value is less than or equal to the deftype specified value of timeout when the autonomic quiescing of unresponsive name servers function is active

Explanation
Check CSRES_AUTOQ_TIMEOUT ran successfully and found no exceptions. The check detected that the resolver timeout value is less than or equal to the timeout value specified for the check when the autonomic quiescing of unresponsive name servers function is active. This function is enabled by specifying the AUTOQUIESCE operand on the UNRESPONSIVETHRESHOLD resolver setup statement. The resolver timeout value is specified by using the RESOLVERTIMEOUT statement in the global TCPIP.DATA file. The default value is 5 seconds.

In the message text:

deftype
Indicates whether the maximum value for this check has been overridden by the user.

OWNER
The default maximum check value has not been overridden by the user.

INSTALLATION
The maximum check value has been overridden by the user.

timeout
The maximum check value against which the resolver timeout value is compared.

System action
The system continues processing.
**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBHCC2

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**

EZBH017I - The resolver timeout value is less than or equal to the installation specified value of 5 when the autonomic quiescing of unresponsive name servers function is active

EZBH018E  The resolver timeout value is greater than the deftype specified value of timeout when the autonomic quiescing of unresponsive name servers function is active.

**Explanation**
Check CSRES_AUTOQ_TIMEOUT detected that the RESOLVETIMEOUT statement is specified with a value greater than the timeout value specified for the check when the autonomic quiescing of unresponsive name servers function is active. This function is enabled by specifying the AUTOQUIESCE operand on the UNRESPONSIVETHRESHOLD resolver setup statement.

The default value for the RESOLVETIMEOUT statement is 5 seconds. When the autonomic quiescing of unresponsive name servers function is active, the resolver polls unresponsive name servers every six seconds. The resolver uses the value from the RESOLVETIMEOUT statement in the global TCPIP.DATA file, or the default value, to determine how long to wait for a response to the poll. If you specify a value for the RESOLVETIMEOUT statement that is greater than 5 seconds, the resolver will be less efficient when polling unresponsive name servers.

In the message text:

**deftype**
Indicates whether the maximum value for this check has been overridden by the user.
OWNER
The default maximum check value has not been overridden by the user.

INSTALLATION
The maximum check value has been overridden by the user.

timeout
The maximum check value against which the resolver timeout value is compared.

System action
The system continues processing.

Operator response
If you want to continue using the autonomic quiescing function with the current RESOLVERTIMEOUT value, no action is needed.
If you no longer want to use the autonomic quiescing function, or the RESOLVERTIMEOUT value must be less than or equal to the timeout value, contact the system programmer.

System programmer response
If you want to use the autonomic quiescing function with the current RESOLVERTIMEOUT value, no action is needed.
If you want to use the autonomic quiescing function with a RESOLVERTIMEOUT value less than or equal to the timeout value, change the value on the RESOLVERTIMEOUT statement in the global TCPIP.DATA file. After the global TCPIP.DATA file has been corrected, instruct the operator to issue a MODIFY RESOLVER,REFRESH command to update the value.
If you no longer want to use the autonomic quiescing function, remove the AUTOQUIESCE operand from the UNRESPONSIVETHRESHOLD resolver setup statement in the resolver setup file. After the resolver setup file has been corrected, instruct the operator to issue a MODIFY RESOLVER,REFRESH,SETUP=setup_file_name command to stop the autonomic quiescing function.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBHCCK2

Routing code
Not applicable.

Descriptor code
3

Automation
This message is a candidate for automation in cases where different global TCPIP.DATA files exist.
Example

EZBH018E - The resolver timeout value is greater than the installation specified value of 5 when the autonomic quiescing of unresponsive name servers function is active

EZBH019I - This check is not applicable when the autonomic quiescing of unresponsive name servers function is not active

Explanation

Check CSRES_AUTOQ_TIMEOUT is not applicable when the autonomic quiescing of unresponsive name servers function is not active. This function is enabled by specifying the AUTOQUIESCE operand on the UNRESPONSIVETHRESHOLD resolver setup statement.

System action

The system continues processing.

Operator response

None.

System programmer response

None.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP

Module

EZBHCCK2

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

This message is not a candidate for automation.

Example

EZBH019I - This check is not applicable when the autonomic quiescing of unresponsive name servers function is not active

EZBH020I - The resolver is using UDP to communicate with name servers when the autonomic quiescing of unresponsive name servers function is active.
Explanation
Check CSRES_AUTOQ_RESOLVEVIA ran successfully and found no exceptions. The check detected that the resolver is using UDP to communicate with the name server when the autonomic quiescing of unresponsive name servers function is active. This function is enabled by specifying the AUTOQUIESCE operand on the UNRESPONSIVETHRESHOLD resolver setup statement. The protocol used by the resolver to communicate with name servers is specified by using the RESOLVEVIA statement in the global TCPIP.DATA file. The default value is UDP.

System action
The system continues processing.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBHCK2

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
This message is not a candidate for automation.

Example
EZBH020I - The resolver is using UDP to communicate with name servers when the autonomic quiescing of unresponsive name servers function is active.

EZBH021E - The resolver is using TCP to communicate with name servers when the autonomic quiescing of unresponsive name servers function is active.

Explanation
Check CSRES_AUTOQ_RESOLVEVIA detected that the RESOLVEVIA statement is specified with the value TCP in the global TCPIP.DATA file when the autonomic quiescing of unresponsive name servers function is active. This
function is enabled by specifying the AUTOQUIESCE operand on the UNRESPONSIVETHRESHOLD resolver setup statement.

The default value for the RESOLVEVIA statement is UDP. If the autonomic quiescing of unresponsive name server function is active, the resolver polls unresponsive name servers by using UDP, not TCP. If your installation uses TCP to forward the DNS queries generated by an application to name servers, the results of the resolver polling attempts by using UDP will not accurately reflect the responsiveness of the name server in your installation.

**System action**
The system continues processing.

**Operator response**
If you want to continue using the autonomic quiescing function with RESOLVEVIA TCP, no action is needed.
If you no longer want to use the autonomic quiescing function, or the RESOLVEVIA value should be UDP, contact the system programmer.

**System programmer response**
If you want to use the autonomic quiescing function with RESOLVEVIA TCP, no action is needed.
If you want to use the autonomic quiescing function with RESOLVEVIA UDP, remove the RESOLVEVIA statement or change the value to UDP in the global TCPIP.DATA file. After the global TCPIP.DATA file is corrected, instruct the operator to issue a MODIFY RESOLVER,REFRESH command to update the value.
If you no longer want to use the autonomic quiescing function, remove the AUTOQUIESCE operand from the UNRESPONSIVETHRESHOLD resolver setup statement in the resolver setup file. After the resolver setup file has been corrected, instruct the operator to issue a MODIFY RESOLVER,REFRESH,SETUP=setup_file_name command to stop the autonomic quiescing function.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBHCK2

**Routing code**
Not applicable.

**Descriptor code**
3

**Automation**
This message is a candidate for automation in cases where different global TCPIP.DATA files exist.
Example

EZBH021E - The resolver is using TCP to communicate with name servers when the autonomic quiescing of unresponsive name servers function is active

EZBH022I - This check is not applicable when the autonomic quiescing of unresponsive name servers function is not active.

Explanation

Check CSRES_AUTOQ_RESOLVEVIA is not applicable when the autonomic quiescing of unresponsive name servers function is not active. This function is enabled by specifying the AUTOQUIESCE operand on the UNRESPONSIVETHRESHOLD resolver setup statement.

System action

The system continues processing.

Operator response

None.

System programmer response

None.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP

Module

EZBHCC2

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

This message is not a candidate for automation.

Example

EZBH022I - This check is not applicable when the autonomic quiescing of unresponsive name servers function is not active.

EZBH900I - INTERNAL ERROR: Function func_name RC return_code RSN reason_code
Explanation
Check processing for the TCP/IP check identified in the preceding IBM Health Checker for z/OS message failed as the result of an internal processing error.

The check name includes the job name of the TCP/IP stack as a suffix.

In the message text:

**func_name**
The name of the function that failed.

**return_code**
The failure return code that was returned by the function. It is displayed as 8 hexadecimal digits. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

**reason_code**
The failure reason code that was returned by the function. It is displayed as 8 hexadecimal digits. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
System processing continues. The check did not run for the specified stack.

Operator response
Contact the system programmer.

System programmer response
Use the return code and reason code values for the function specified to determine the cause of the failure and correct the problem. If the problem cannot be determined, or corrected, contact the IBM Software Support Center.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBHCCK1

Example
EZBH900I INTERNAL ERROR: Function BPX1IOC RC 000003F3 RSN 00000532
Chapter 11. EZD0xxxx messages

**EZD0001I**  SETTING VLAN ID NOT SUPPORTED FOR DEVICE *device_name*

**Explanation**
TCP/IP could not set the VLAN ID for this device. The OSA-Express adapter microcode level does not support setting the VLAN ID.

*device_name* is the name of the device.

**System action**
TCP/IP prevents the device from activating.

**Operator response**
Inform the system programmer about the error.

**System programmer response**
Install a level of OSA-Express microcode that supports the VLAN ID function. Use the VTAM DISPLAY TRL command to determine the OSA-Express microcode level. For more information about the VTAM DISPLAY TRL command, see z/OS Communications Server: SNA Operation.

**Module**
TCPIP

**Procedure name**
EZBIFIND

**EZD0002I**  ERROR SETTING VLAN ID FOR DEVICE *device_name*

**Explanation**
An unexpected error occurred while setting the VLAN ID for the device.

*device_name* is the name of the device.

**System action**
TCP/IP prevents the device from activating.

**Operator response**
Inform the system programmer about the error.

**System programmer response**
Obtain a TCP/IP CTRACE with the VTAM option and contact IBM software support services.

**Module**
TCPIP
EZD0003I  SETTING VLAN ID NOT SUPPORTED FOR INTERFACE interface_name

Explanation
TCP/IP could not set the VLAN ID for this interface. The OSA-Express adapter microcode level does not support setting the VLAN ID.
interface_name is the name of the interface.

System action
TCP/IP prevents the device from activating.

Operator response
Inform the system programmer about the error.

System programmer response
Install a level of OSA-Express microcode that supports the VLAN ID function. Use the VTAM DISPLAY TRL command to determine the OSA-Express microcode level. For more information about the VTAM DISPLAY TRL command, see z/OS Communications Server: SNA Operation.

Module
TCPIP

EZD0004I  ERROR SETTING VLAN ID FOR INTERFACE interface_name

Explanation
An unexpected error occurred while setting the VLAN ID for the interface.
If the interface is configured with CHPID type OSX, one possible reason for this error is that the VLAN ID is not authorized for the ensemble virtual network.
interface_name is the name of the interface.

System action
TCP/IP prevents the interface from activating.

Operator response
Inform the system programmer about the error.

System programmer response
Obtain a TCP/IP CTRACE with the VTAM option and contact IBM software support services.
If the interface is configured with CHPID type OSX, ensure that the virtual network you created through the network virtualization function specifies the same VLAN ID as the OSX interface.

Module
TCPIP
**EZD0005I**  SETTING VLAN USER PRIORITY NOT SUPPORTED FOR INTERFACE *interface_name*

**Explanation**
TCP/IP cannot use VLAN priority for this interface. The OSA-Express adapter microcode level does not support the VLAN priority function.

*interface_name* is the name of the interface.

**System action**
TCP/IP continues with interface activation but does not use VLAN priority.

**Operator response**
Inform the system programmer about the error.

**System programmer response**
Install a level of OSA-Express microcode that supports the VLAN priority function. Use the VTAM DISPLAY TRL command to determine the OSA-Express microcode level. For more information about the VTAM DISPLAY TRL command, see z/OS Communications Server: SNA Operation.

**Module**
TCPIP

**EZD0006I**  ERROR SETTING VLAN USER PRIORITY FOR INTERFACE *interface_name*

**Explanation**
An unexpected error occurred while setting the VLAN priority for the interface.

*interface_name* is the name of the interface.

**System action**
TCP/IP continues with interface activation but does not use VLAN priority.

**Operator response**
Inform the system programmer about the error.

**System programmer response**
Obtain a TCP/IP CTRACE with the VTAM option and contact IBM software support services.

**Module**
TCPIP
EZD0007I CONNECTION TO addr CLEARED FOR INTERFACE Interface_Name

Explanation
TCP/IP was notified that the MPCPTP6 connection to the specified IPv6 address for the specified interface is no longer active.

addr is the IPv6 address of the remote node.

Interface_Name is the name of the interface.

System action
The specified connection is no longer available for use with TCP/IP. TCP/IP will attempt to recover the connection.

Operator response
None.

System programmer response
None.

Module
EZBIFIUM

Procedure name
CM_CLEAR_IND_ATMMPC

EZD0008I TIMEOUT DURING MPCPTP OR MPCPTP6 CONTROL PACKET EXCHANGE — DEACTIVATING dev_or_interface_name

Explanation
The remote node is not responding during the MPCPTP or MPCPTP6 control packet exchange.

dev_or_interface_name is the name of the device or interface. This value of this field displays in the format of DEVICE device_name or INTERFACE interface_name.

System action
The IPv6 Interface or IPv4 Link on which the timeout occurred is deactivated.

Operator response
Contact the system programmer.

System programmer response
Obtain a CTRACE with OPTIONS=VTAM,VTAMDATA (on both sides, if the remote node is an IBM z/OS host) and contact the IBM Software Support Center. If the remote node is not an IBM z/OS host, the IBM Software Support Center might instruct you to contact the service group for the remote node.

Module
EZBIFOUT
**Procedure name**
Process_MPC_Rxmt_Queues

**EZD0009I CONNECTION TO addr ACTIVE FOR INTERFACE interface_name**

**Explanation**
The MPCPTP6 connection to the specified IPv6 address for the specified interface is now active.

*addr* is the IPv6 address of the remote node.

*interface_name* is the name of the interface.

**System action**
The specified connection is now available for use by TCP/IP.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZBIFIUM

**Procedure name**
DM_ACT_IND_MPC

**EZD0010I ERROR: CODE= Error_Code DURING Link_Control_Function CONNECTION TO addr FOR INTERFACE Interface_Name DIAGNOSTIC CODE: Internal_Diagnostic_Code**

**Explanation**
The Link Layer has detected an error during activation of a virtual connection (VC) for an MPCPTP6 interface.

*Error_Code* is the Data Link Control (DLC) Status Code for the link layer.

*Link_Control_Function* is the function being performed on the VC.

*addr* is the IPv6 address of the remote node.

*Interface_Name* is the name of the interface.

*Internal_Diagnostic_Code* is an internal diagnostic code for use by IBM.

**System action**
TCP/IP does not activate the VC.

**Operator response**
Consult the section on Data Link Control (DLC) Status Codes in z/OS Communications Server: IP and SNA Codes for a description of the status code for the link layer. If a hardware problem is indicated, correct the hardware problem and restart the MPCPTP6 interface.

**System programmer response**
Perform the action described in the z/OS Communications Server: IP and SNA Codes for the indicated status code.
EZD0012I DUPLICATE ADDRESS DETECTION PREVENTED BY IPSEC FOR type CONFIGURED HOME ADDRESS addr ON INTERFACE intfname

Explanation
TCP/IP cannot verify the uniqueness of this IP address because the IPSec policy is preventing the stack from sending a neighbor solicitation packet for duplicate address detection.

In the message text:

- **type**
  - Specifies how the home address was configured. Possible values are:
  - **MANUAL**
    - Indicates that the home address was manually configured in the TCP/IP profile.
  - **AUTO**
    - Indicates that the address is either a link-local automatically generated address or an automatically configured address.

- **addr**
  - Specifies the address for which TCP/IP was attempting duplicate address detection.

- **intfname**
  - The interface name.

System action
TCP/IP marks the home address as unavailable for use. If the home address is the link-local address, then TCP/IP deactivates the interface.

Operator response
Contact the system programmer.

System programmer response
Configure filter rules to permit neighbor solicitation and configure neighbor advertisement packets to enable duplicate address detection.

See the following in z/OS Communications Server: IP Configuration Guide:
- The information about special considerations when using IP security for IPv6 for more information about configuring IP filter rules
- The information about setting up physical characteristics in PROFILE.TCPIP for example filter rules
- The Policy sample files for example filter rules

User response
Not applicable.

Problem determination
Not applicable.
**Module**
EJB6PDAD

**Example**
None.

**EZD0013I**  LINK `takeback_link_name` HAS TAKEN BACK ARP RESPONSIBILITY FROM LINK `takeover_link_name`

**Explanation**
The link specified by the `takeback_link_name` value became inactive and the link specified by the `takeover_link_name` value took over the ARP responsibility. The takeback link became active again, so it has taken back the ARP responsibility.

In the message text:
- **`takeback_link_name`**
  - The name of the link that has taken back the ARP responsibility.
- **`takeover_link_name`**
  - The name of the link that has released the ARP responsibility.

**System action**
TCP/IP reassigns ARP responsibility to the takeback link. TCP/IP sends a gratuitous ARP for the IP address of the takeback link that will reply to ARP requests as the owner.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Module**
EJBIFSTC

**Example**
None.

**EZD0014I**  INTERFACE `takeback_interface_name` HAS TAKEN BACK ND RESPONSIBILITY FROM INTERFACE `takeover_interface_name`

**Explanation**
The interface specified by the `takeback_interface_name` value became inactive and the interface specified by the `takeover_interface_name` value took over the neighbor discovery (ND) address resolution responsibility. The takeback interface is active again, so it has taken back the ND address resolution responsibility.

In the message text:
- **`takeback_interface_name`**
  - The name of the interface that has taken back the ND responsibility.
- **`takeover_interface_name`**
  - The name of the interface that has released the ND responsibility.
**takeback_interface_name**

The name of the interface that has taken back the ND Address Resolution responsibility.

**takeover_interface_name**

The name of the interface that has released the ND address resolution responsibility.

**System action**

TCP/IP reassigns ND address resolution responsibility to the takeback interface. TCP/IP sends a gratuitous neighbor advertisement for the IP address of the takeback interface that will reply to neighbor solicitation requests as the owner.

**Operator response**

None.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Module**

EZBIFSTC

**Example**

None.

---

**EZD0015I  osa_portname DOES NOT SUPPORT OSAENTA TRACE**

**Explanation**

The VARY TCPIP,,OSAENTA command or an OSAENTA statement in PROFILE.TCPIP was specified with the ON parameter for an OSA-Express, which does not support the OSA-Express network traffic analyzer trace function.

In the message text:

*osa_portname*

The value specified on the PORTNAME= parameter on the OSAENTA command or statement.

**System action**

TCP/IP does not activate the OSAENTA trace function on this OSA-Express.

**Operator response**

Contact the system programmer.

**System programmer response**

Install a level of OSA-Express microcode that supports the OSAENTA trace function. Use the VTAM DISPLAY TRL command to determine the currently installed OSA-Express Licensed Internal Code (LIC) level. See the information about the DISPLAY TRL command in z/OS Communications Server: SNA Operation.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: TCP/IP stack

Module
EZBIFIND

Routing code
2,8

Descriptor code
12

Example
Not applicable.

EZD0016I  OSAENTA TRACE ENABLED FOR osa_portname

Explanation
TCP/IP successfully activated the OSA-Express network traffic analyzer trace function with the OSA-Express adapter in response to a VARY TCPIP,,OSAENTA command or OSAENTA statement in PROFILE.TCPIP with the ON parameter.

In the message text:

osa_portname
The value specified on the PORTNAME= parameter, which was specified on the OSAENTA command or statement.

System action
TCP/IP starts collecting trace data from the OSA-Express according to the current OSAENTA settings.

Operator response
No action needed, but if you want to display the current OSAENTA settings, use the Netstat DEvlinks/-d command. See the information about Netstat in z/OS Communications Server: IP System Administrator's Commands.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.
EZD0017I  OSAENTA TRACE MODIFIED FOR osa_portname

Explanation
TCP/IP successfully modified the OSA-Express network traffic analyzer trace settings with the OSA-Express adapter in response to a VARY TCPIP,,OSAENTA command or OSAENTA statement in PROFILE.TCPIP.

In the message text:
osal_portname
  The value specified on the PORTNAME= parameter, which was specified on the OSAENTA command or statement.

System action
TCP/IP starts collecting trace data from the OSA-Express according to the new OSAENTA settings.

Operator response
No action needed, but if you want to display the current OSAENTA settings, use the Netstat DEVLINKS/-d command. See the information about NETSTAT in z/OS Communications Server: IP System Administrator's Commands

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.
**Routing code**
2,8

**Descriptor code**
12

**Example**
Not applicable.

**EZD0018I** OSAENTA TRACE DISABLED FOR *osa_portname*

**Explanation**
TCP/IP has deactivated the OSA-Express network traffic analyzer trace function in response to a V TCP/IP,,OSAENTA command or OSAENTA statement in PROFILE.TCPIP with the OFF parameter.

In the message text:

*osa_portname*
The value specified on the PORTNAME= parameter, which was specified on the OSAENTA command or statement.

**System action**
TCP/IP stops collecting trace data on this OSA-Express.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: TCP/IP stack

**Module**
EZBIFIND

**Routing code**
2,8

**Descriptor code**
12
Example
Not applicable.

**EZD0019I** OSAENTA TRACE STOPPED FOR *osa_portname* - REASON: *reason* LIMIT REACHED

**Explanation**
The stack reached one of the limits specified on the V TCPIP,,OSAENTA command or the OSAENTA statement in PROFILE.TCPIP for OSA-Express network traffic analyzer trace collection, and stopped collecting trace data on this OSA-Express.

In the message text:

*osa_portname*  
The value specified on the PORTNAME= parameter, which was specified on the OSAENTA command or statement.

*reason*  
The specific limit that was reached. The limit was one of the following:

- DATA - amount of data traced
- FRAMES - number of frames traced
- RECORD - number of records traced
- TIME - length of time the trace was active

**System action**
TCP/IP stops collecting trace data on this OSA-Express. If the trace is restarted, the current OSAENTA settings remain in effect.

**Operator response**
Contact the system programmer.

**System programmer response**
To restart the trace, issue a VARY TCPIP,,OSAENTA command specifying the ON parameter or issue a VARY TCPIP,,OBEYFILE command with a TCP/IP profile that contains an OSAENTA statement specifying the ON parameter. See the information about the VARY TCPIP,,OSAENTA command in *z/OS Communications Server: IP System Administrator's Commands* and the OSAENTA statement in *z/OS Communications Server: IP Configuration Reference* for more information.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: TCP/IP stack

**Module**
EZBIFIND

**Routing code**
2,8
**EZD0020I**  ERROR *error_code* ENABLING OSAENTA TRACE FOR *osa_portname*

**Explanation**

TCP/IP attempted to enable the OSA-Express network traffic analyzer trace in response to a VARY TCPIP,,OSAENTA command or OSAENTA statement in PROFILE.TCPIP with the ON parameter. However, OSA-Express reported an error that prevents the stack from enabling the trace.

In the message text:

*error_code*

The error code returned by the OSA-Express.

*osa_portname*

The value specified on the PORTNAME= parameter, which was specified on the OSAENTA command or statement.

**System action**

TCP/IP does not enable the OSAENTA trace on this OSA-Express.

**Operator response**

Contact the system programmer.

**System programmer response**

See OSA Network Traffic Analyzer (OSAENTA) Error Codes in z Systems: Open Systems Adapter-Express Customer's Guide and Reference for a description of the error code and its potential action. After correcting the error, use the OSAENTA command or statement to activate the trace. See the information about the VARY TCPIP,,OSAENTA command in z/OS Communications Server: IP System Administrator’s Commands and the OSAENTA statement in z/OS Communications Server: IP Configuration Reference for more information.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: TCP/IP stack

**Module**

EZBIFIND

**Routing code**

2,8
**EZD0021I** ERROR *error_code* MODIFYING OSAENTA TRACE FOR *osa_portname*

**Explanation**

TCP/IP attempted to modify the OSA-Express network traffic analyzer trace settings with the OSA-Express adapter in response to a VARY TCPIP,,OSAENTA command or OSAENTA statement in PROFILE.TCPIP. However, OSA-Express reported an error that prevents the stack from modifying the trace settings.

In the message text:

*error_code*

The error code returned by the OSA-Express.

*osa_portname*

The value specified on the PORTNAME= parameter, which was specified on the OSAENTA command or statement.

**System action**

TCP/IP disables the OSAENTA trace on this OSA-Express.

**Operator response**

Contact the system programmer.

**System programmer response**

See OSA Network Traffic Analyzer (OSAENTA) Error Codes in z Systems: Open Systems Adapter-Express Customer’s Guide and Reference for information about OSA reject codes and a description of the error code and its potential action. After correcting the error, use the OSAENTA command or statement to activate the trace. See the information about the VARY TCPIP,,OSAENTA command in z/OS Communications Server: IP System Administrator’s Commands and the OSAENTA statement in z/OS Communications Server: IP Configuration Reference for more information.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: TCP/IP stack

**Module**

EZBIFIND

**Routing code**

2,8
EZD0022I  INTERFACE interface_name DOES NOT SUPPORT THE ISOLATE FUNCTION

Explanation
The ISOLATE parameter was specified on the INTERFACE statement for this device but the level of OSA-Express adapter microcode that is installed does not support the ISOLATE function.

In the message text:

interface_name
The name of the interface with the ISOLATE parameter specified.

System action
Interface activation fails.

Operator response
Contact the system programmer.

System programmer response
Either install a level of OSA-Express microcode that supports the ISOLATE function or remove the ISOLATE parameter from the INTERFACE statement. Use the VTAM DISPLAY TRL command to determine what level of the OSA-Express Licensed Internal Code (LIC) is currently installed. See the information about the DISPLAY TRL command information in z/OS Communications Server: SNA Operation for more information.

The ISOLATE function is limited to OSA-Express2 or OSA-Express3 ethernet features in QDIO mode and running at least an IBM System z9® Enterprise Class (EC) or z9 Business Class (BC). See the 2094, 2096, 2097, and 2098 DEVICE Preventive Service Planning (PSP) buckets for more information.

See the information about the INTERFACE–IPAQENET OSA-Express QDIO interface statement and the INTERFACE–IPAQENET6 OSA-Express QDIO interface statement in z/OS Communications Server: IP Configuration Reference for information about OSA-Express connection isolation.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Configuration & Initialization

Module
EZBIFIND

Routing code
2,8
Descriptor code
12

Automation
This message is displayed on the system console. Automation might be appropriate because the activation of the interface is not allowed when this message is issued.

Example

EZD0022I INTERFACE OSA3 DOES NOT SUPPORT THE ISOLATE FUNCTION.

EZD0023I IPV6 MULTIPATH PERPACKET NOT VALID WITH IPV6 SECURITY – IPV6 MULTIPATH SUPPORT DISABLED FOR ROUTE TABLE table

Explanation
IPv6 multipath perpacket cannot be enabled for a policy-based route table when IPv6 security support is enabled on the TCP/IP stack.

In the message text:

table
The name of a policy-based route table

System action
TCP/IP continues. IPv6 multipath support is disabled for the specified route table.

Operator response
Contact the system programmer.

System programmer response
To use IPv6 multipath support with IPv6 security, enable IPv6 multipath per connection support by coding PerConnection on the Multipath6 parameter of the RouteTable statement in the policy configuration. See policy-based routing (Routing) policy statements in z/OS Communications Server: IP Configuration Reference for information about the RouteTable policy statement.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZB6PPBR

Routing code
2
Automation
This message is displayed on the console. Automation can detect when IPv6 multipath perpacket is configured for a policy-based route table and IPv6 security support is enabled on the TCP/IP stack.

Example

```
EZD0023I IPV6 MULTIPATH PERPACKET NOT VALID WITH IPV6 SECURITY
   - IPV6 MULTIPATH SUPPORT DISABLED FOR ROUTE TABLE FTPRTE56

EZD0024I DEVICE device_name DOES NOT SUPPORT VMAC
```

Explanation
Virtual MAC (VMAC) was specified on the LINK statement for this device but the OSA-Express adapter microcode level does not support assigning Virtual MAC addresses. TCP/IP could not assign a VMAC address for this device during device activation.

In the message text:

`device_name`
The name of the device with VMAC configured.

System action
Device activation fails.

Operator response
Contact the system programmer.

System programmer response
Either install a level of OSA-Express microcode that supports the virtual MAC function or remove the VMAC parameter from the LINK statement. Use the VTAM DISPLAY TRL command to determine the currently installed OSA-Express Licensed Internal Code (LIC) level. See the information about the DISPLAY TRL command in z/OS Communications Server: SNA Operation for more information. See the 2094DEVICE Preventive Service Planning (PSP) bucket and the 2096DEVICE Preventive Service Planning (PSP) bucket for the OSA-Express LIC levels required for VMAC support. See the information about OSA-Express Virtual MAC (VMAC) routing or workload balancing in z/OS Communications Server: IP Configuration Guide for information about the networking affects of configuring with and without VMAC addresses.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Storage Utilization

Module
EZBIFIND
Routing code
2,8

Descriptor code
12

Example
Not applicable.

EZD0025I  INTERFACE interface_name DOES NOT SUPPORT VMAC

Explanation
Virtual MAC (VMAC) was specified on the INTERFACE statement for this device but the OSA-Express adapter microcode level does not support assigning VMAC addresses. TCP/IP could not assign a VMAC address for this interface during interface activation.

In the message text:

interface_name
The name of the interface with VMAC configured.

System action
Interface activation fails.

Operator response
Contact the system programmer.

System programmer response
Either install a level of OSA-Express microcode that supports the Virtual MAC function or remove the VMAC parameter from the LINK statement. Use the VTAM DISPLAY TRL command to determine the currently installed OSA-Express Licensed Internal Code (LIC) level. See the information about the DISPLAY TRL command in z/OS Communications Server: SNA Operation for more information. See the 2094DEVICE Preventive Service Planning (PSP) bucket and the 2096DEVICE Preventive Service Planning (PSP) bucket for the OSA-Express LIC levels required for VMAC support. See the information about OSA-Express Virtual MAC (VMAC) routing or workload balancing in z/OS Communications Server: IP Configuration Guide for information about the networking affects of configuring with and without VMAC addresses.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Stack Configuration

Module
EZBIFIND
Routing code
2,8

Descriptor code
12

Example
Not applicable.

EZD0026I  ERROR error_code ASSIGNING VMAC TO DEVICE device_name

Explanation
An unexpected error occurred while assigning a virtual MAC (VMAC) address to the device.

In the message text:

error_code
The error code returned from the OSA-Express microcode.

device_name
The name of the device.

System action
Device activation fails.

Operator response
Contact the system programmer.

System programmer response
See OSA Reject Codes and Internal Errors in z Systems: Open Systems Adapter-Express Customer's Guide and Reference for information about the OSA Reject Codes and a description of the error.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Stack Configuration

Module
EZBIFIND

Routing code
2,8

Descriptor code
12
Example
Not applicable.

**EZW0027I**  ERROR *error_code* ASSIGNING VMAC TO INTERFACE *interface_name*

**Explanation**
An unexpected error occurred while assigning a virtual MAC (VMAC) address to the interface.

In the message text:

*error_code*
The error code returned from the OSA-Express microcode.

*interface_name*
The name of the interface.

**System action**
Interface activation fails.

**Operator response**
Contact the system programmer.

**System programmer response**
See OSA Reject Codes and Internal Errors in *z Systems: Open Systems Adapter-Express Customer’s Guide and Reference* for information about the OSA Reject Codes and a description of the error.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: Stack Configuration

**Module**
EZBIFIND

**Routing code**
2,8

**Descriptor code**
12

**Example**
Not applicable.

**EZW0028I**  IPV4 MULTIPATH PERPACKET NOT VALID WITH IPV4 SECURITY - MULTIPATH SUPPORT DISABLED FOR ROUTE TABLE *table*
Explanation
IPv4 multipath perpacket cannot be enabled for a policy-based route table when IPv4 security support is enabled on the TCP/IP stack.

In the message text:

**table**
- The name of a policy-based route table.

**System action**
TCP/IP continues. IPv4 multipath support is disabled for the specified route table.

**Operator response**
Contact the system programmer.

**System programmer response**
If you want to use IPv4 multipath support in conjunction with IPv4 security, enable multipath per connection support by coding **PerConnection** on the Multipath parameter of the RouteTable statement in the policy configuration. See the information about the policy-based routing (Routing) policy statements in z/OS Communications Server: IP Configuration Reference for information about the RouteTable policy statement.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBIPPBR

**Routing code**
2

**Descriptor code**
12

**Example**

```
EZD0028I IPV4 MULTIPATH PERPACKET NOT VALID WITH IPV4 SECURITY - MULTIPATH SUPPORT DISABLED FOR ROUTE TABLE FTPRTES
```

**EZD0029I** PATH MTU DISCOVERY SUPPORT IS DISABLED FOR ROUTE TABLE table

**Explanation**
Path MTU discovery support was disabled for the specified policy-based route table. This occurred because a value of Yes was coded for the IgnorePathMtuUpdate parameter on the RouteTable policy statement that defined the route table.

In the message text:
**table**
The name of a policy-based route table.

**System action**
TCP/IP continues. Path MTU discovery updates will not be applied to any routes in the specified route table.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBIPPB

**Routing code**
2

**Descriptor code**
12

**Example**

EZD0029I PATH MTU DISCOVERY SUPPORT IS DISABLED FOR ROUTE TABLE FTPRTES

EZD0030I DISPLAY OSAINFO FAILED FOR name - reason

**Explanation**
The DISPLAY TCPIP,,OSAINFO request could not be processed for one of the reasons listed below.

In the message text:

**name**
The INTFNAME value specified on the DISPLAY TCPIP,,OSAINFO command.

**reason**
The reason for the error. Possible values are:

**LINK OR INTERFACE DOES NOT EXIST**
The LINK or INTERFACE name does not exist.

**LINK OR INTERFACE DOES NOT SUPPORT DISPLAY OSAINFO**
The LINK or INTERFACE does not support the DISPLAY TCPIP,,OSAINFO command.
**LINK OR INTERFACE NOT ACTIVE**  
The OSA-Express is not currently active.

**COMMAND ALREADY IN PROGRESS**  
A DISPLAY TCPIP,,OSAINFO command is already in progress for the OSA-Express.

**STORAGE NOT AVAILABLE**  
Storage required to complete the DISPLAY TCPIP,,OSAINFO request could not be obtained.

**COMMAND TIMED OUT**  
The OSA-Express has taken too long to reply with the information.

**System action**  
Processing of the DISPLAY TCPIP,,OSAINFO command is discontinued.

**Operator response**  
Not applicable.

**System programmer response**  
For persistent COMMAND TIMED OUT failures, obtain a TCP/IP CTRACE with the VTAM option and contact IBM software support services.

**User response**  
Follow the procedure described in the Problem Determination and re-submit the DISPLAY TCPIP,,OSAINFO command.

**Problem determination**
- If the reason value is LINK OR INTERFACE DOES NOT EXIST, issue the DISPLAY TCPIP,,NETSTAT,DEVLINKS
- If the reason value is LINK OR INTERFACE NOT ACTIVE, start the LINK or INTERFACE.
- If the reason value is LINK OR INTERFACE DOES NOT SUPPORT DISPLAY OSAINFO, either the LINK or INTERFACE is not OSA-Express or the OSA-Express must be upgraded prior to issuing this command. DISPLAY TCPIP,,OSAINFO is limited to OSA-Express3 ethernet features in QDIO mode (CHPID type OSD, OSM, or OSX).
- If the reason value is COMMAND ALREADY IN PROGRESS, wait for the previous command to complete.
- If the reason value is STORAGE NOT AVAILABLE, determine and fix the reason for the storage shortage. See the information about the diagnosing storage abends and storage growth in z/OS Communications Server: IP Diagnosis Guide for more information about storage problems.
- If the reason value is COMMAND TIMED OUT, the OSA-Express is either extremely busy or an internal error occurred. If the failure persists contact the system programmer.

**Source**  
z/OS Communications Server TCP/IP

**Module**  
EZBIFIND, EZBIFDOB

**Routing code**  
*

**Descriptor code**  
*
Automation
Not applicable.

Example

EZD0030I  DISPLAY OSAINFO FAILED FOR NSQDIO11 - LINK OR INTERFACE DOES NOT EXIST

EZD0031I  TCP/IP CS versionRelease TCPIP Name: name time

Explanation
This is the first message in the DISPLAY TCPIP,OSAINFO command report. See the information about the DISPLAY TCPIP,OSAINFO command in z/OS Communications Server: IP System Administrator's Commands for a detailed description of the report.

EZD0032I  IPV6 PATH MTU DISCOVERY SUPPORT IS DISABLED FOR ROUTE TABLE table

Explanation
IPv6 path MTU discovery support was disabled for the specified policy-based route table. This occurred because a value of Yes was coded for the IgnorePathMtuUpdate6 parameter on the RouteTable policy statement that defined the route table.

In the message text:

*table*
   The name of a policy-based route table

System action
TCP/IP continues. IPv6 path MTU discovery updates will not be applied to any routes in the specified route table.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZB6PPBR

Routing code
2
**Automation**

This message is displayed on the console. Automation can detect when IPv6 path MTU discovery support is disabled for a policy-based route table.

**Example**

```
EZD0032I  IPV6 PATH MTU DISCOVERY SUPPORT IS DISABLED FOR ROUTE TABLE FTPRTES6
EZD0033I  GATEWAY ADDRESS ipaddress SPECIFIED IN ROUTE TABLE rttable IS NOT VALID
```

**Explanation**

A route entry or dynamic routing parameter entry in the specified policy-based route table contains a gateway address that is not valid. This message might be issued because the gateway address is a local address.

In the message text:

- `ipaddress`
  The gateway address specified on a route entry or dynamic routing parameter entry in a RouteTable statement defined in a Policy Agent configuration file.

- `rttable`
  The name of the route table in a Policy Agent configuration file.

**System action**

Processing continues. The route is not added to the policy-based route table.

**Operator response**

Contact the system programmer.

**System programmer response**

Correct the entry in the RouteTable statement in the Policy Agent configuration file. See the information about the policy-based routing (Routing) policy statements in z/OS Communications Server: IP Configuration Reference for information about configuring a route entry or dynamic routing parameters entry.

**User response**

Not applicable.

**Problem determination**

See the system programmer response.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZBIPRTE, EZBIPPBR, EZB6PRTE, EZB6PPBR

**Routing code**

2
EZD0033I GATEWAY ADDRESS 127.0.0.2 SPECIFIED IN ROUTE TABLE FTPRTES IS NOT VALID
EZD0033I GATEWAY ADDRESS FE80::3 SPECIFIED IN ROUTE TABLE FTPRTES6 IS NOT VALID

EZD0034I INTERFACE interface SPECIFIED IN ROUTE TABLE rttable IS NOT VALID

Explanation
A route entry or dynamic routing parameters entry in the specified policy-based route table contains an interface that is not valid. This message might be issued because the interface is a VIPA or loopback interface.

In the message text:
interface
The link name specified on a route entry or dynamic routing parameters entry in a RouteTable statement that is defined in a Policy Agent configuration file.

rttable
The name of the route table in a Policy Agent configuration file.

System action
Processing continues. The route is not added to the policy-based route table.

Operator response
Contact the system programmer.

System programmer response
Correct the route entry in the RouteTable statement in the Policy Agent configuration file. See the information about the policy-based routing (Routing) policy statements in z/OS Communications Server: IP Configuration Reference for more information about configuring the route entry or dynamic routing parameters entry.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP

Module
EZBIPRTE, EZBIPPB, EZB6PRTE, EZB6PPBR

Routing code
2

Descriptor code
12
Example

EZD0034I INTERFACE VIPA1 SPECIFIED IN ROUTE TABLE FTPRTES IS NOT VALID

EZD0035I DESTINATION ADDRESS ipaddress SPECIFIED IN ROUTE TABLE rtable IS NOT VALID

Explanation
A route entry in the specified policy-based route table contains a destination address that is not valid. This message might be issued because the destination address is a local address.

In the message text:

*ipaddress*

The destination address specified on a route entry in a RouteTable statement that is defined in a Policy Agent configuration file.

*rtable*

The name of the route table in a Policy Agent configuration file.

System action
Processing continues. The route is not added to the policy-based route table.

Operator response
Contact the system programmer.

System programmer response
Correct the route entry in the RouteTable statement in the Policy Agent configuration file. See the information about the policy-based routing (Routing) policy statements in z/OS Communications Server: IP Configuration Reference for more information about configuring the route entry.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP

Module
EZBIPRTE, EZB6PRTE

Routing code
2

Descriptor code
12

Example

EZD0035I DESTINATION ADDRESS 10.1.1.1 SPECIFIED IN ROUTE TABLE FTPRTES IS NOT VALID
EZD0035I DESTINATION ADDRESS 2001:DB8:0:A1B::3:3 SPECIFIED IN ROUTE TABLE FTPRTES6 IS NOT VALID
EZD0036I  DEVICE device_name DOES NOT SUPPORT DYNAMIC INBPERF

Explanation
The value DYNAMIC was specified for the INBPERF parameter on the LINK statement for an OSA-Express adapter. The OSA-Express adapter microcode level does not support the INBPERF DYNAMIC setting.

In the message text:

device_name
    The name of the device from the DEVICE statement.

System action
TCP/IP activates the device with the default INBPERF parameter setting BALANCED.

Operator response
Contact the system programmer.

System programmer response
Install a level of OSA-Express microcode that supports the dynamic LAN idle function. Use the VTAM DISPLAY TRL command to determine your current OSA-Express microcode level.

See the information about the DISPLAY TRL command in z/OS Communications Server: SNA Operation for information about the VTAM DISPLAY TRL command.

See the 2094DEVICE Preventive Service Planning (PSP) and the 2096DEVICE Preventive Service Planning (PSP) buckets for further information about which level of OSA-Express microcode supports the dynamic LAN idle function.

See the information about DEVICE and LINK - MPCIPA OSA-Express QDIO devices information in z/OS Communications Server: IP Configuration Reference for information about the INBPERF parameter.

If you choose to not use the dynamic LAN idle function, remove the DYNAMIC value from the LINK statement.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBIFSTC

Routing code
2,8

Descriptor code
12
EZD0036I DEVICE OSAQDIO1 DOES NOT SUPPORT DYNAMIC INBPERF

EZD0037I INTERFACE interface_name DOES NOT SUPPORT DYNAMIC INBPERF

Explanation
The value DYNAMIC was specified for the INBPERF parameter on the INTERFACE statement for an OSA-Express adapter. The OSA-Express adapter microcode level does not support the INBPERF DYNAMIC setting.

In the message text:

interface_name
The name of the interface from the INTERFACE statement.

System action
TCP/IP activates the associated interface with the default INBPERF parameter setting BALANCED.

Operator response
Contact the system programmer.

System programmer response
Install a level of OSA-Express microcode that supports the dynamic LAN idle function. Use the VTAM DISPLAY TRL command to determine your current OSA-Express microcode level.

See the information about the DISPLAY TRL command in z/OS Communications Server: SNA Operation for information about the VTAM DISPLAY TRL command.

See the 2094DEVICE Preventive Service Planning (PSP) and the 2096DEVICE Preventive Service Planning (PSP) buckets for more information about which level of OSA-Express microcode supports the dynamic LAN idle function.

See the information about INTERFACE - IPAQENET6 OSA-Express QDIO interfaces information in z/OS Communications Server: IP Configuration Reference for information about the INBPERF parameter.

If you choose to not use the dynamic LAN idle function, remove the DYNAMIC value from the INTERFACE statement.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBIFSTC

Routing code
2,8
Example

EZD0037I INTERFACE QDI06101 DOES NOT SUPPORT DYNAMIC INBPERF

EZD0038I ERROR error_code UPDATING DYNAMIC INBPERF SETTINGS FOR osa_portname

Explanation
An unexpected error occurred while dynamically updating the OSA-Express frequency of interruptions for inbound traffic.

In the message text:

error_code
The error code returned from the OSA-Express microcode.

osa_portname
The name of the device or the name of the PORTNAME parameter of the interface that encountered the error.

System action
The OSA-Express adapter will use the last successfully transmitted values to interrupt the host for inbound traffic.

Operator response
Contact the system programmer.

System programmer response
See OSA Reject Codes and Internal Errors in z Systems: Open Systems Adapter-Express Customer’s Guide and Reference for information about the OSA Reject Codes and a description of the error.

If the OSA-Express adapter experiences inbound performance problems, change the INBPERF setting from DYNAMIC to an alternate value on the LINK statement and INTERFACE statement representing this OSA-Express adapter. Then stop and restart the OSA-Express device and interface.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBIFIND

Routing code
2,8
EZD0038I ERROR E00C UPDATING DYNAMIC INBPERF SETTINGS FOR OSAQDIO4

**EZD0039I** INTERFACE *interface_name* IS NOT BROADCAST CAPABLE

**Explanation**
An IPAQENET interface has been activated with the IPBCAST parameter, which indicates that broadcast capability was requested; however, the interface is not broadcast capable.

In the message text:

*interface_name*
  The name of the interface that is not broadcast capable.

**System action**
TCP/IP allows the interface to activate, but broadcast support for this interface will be set to no. No broadcast packets can be sent or received over this interface.

**Operator response**
None.

**System programmer response**
Install the latest level of OSA-Express microcode and restart the interface.

**User response**
Not applicable.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBIFIND

**Routing code**
2,8

**Descriptor code**
12

**Automation**
Not applicable.
Example

EZD0039I INTERFACE OSAQDI04 IS NOT BROADCAST CAPABLE

EZD0040I INTERFACE takeover_interface_name HAS TAKEN OVER ARP RESPONSIBILITY FOR INACTIVE INTERFACE inactive_interface_name

Explanation

An interface became inactive and TCP/IP detected another active interface on the same physical network that can take over ARP responsibility for the inactive interface. If the inactive interface becomes active again, then TCP/IP reassigns the ARP responsibility to that interface. See the information about interface-layer fault-tolerance for local area networks (interface-takeover function) in z/OS Communications Server: IP Configuration Guide for more information about the interface-takeover function.

In the message text:

- **takeover_interface_name**
  - The name of the interface that took over the ARP responsibility.

- **inactive_interface_name**
  - The name of the inactive interface.

System action

TCP/IP assigns ARP responsibility for the inactive interface to the takeover interface. TCP/IP sends a gratuitous ARP for the IP address of the inactive interface and uses the takeover interface to reply to ARP requests on behalf of the inactive interface.

Operator response

None.

System programmer response

None.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP

Module

EZBIFIND

Routing code

2,8

Descriptor code

12
Automation
This message replaces message EZZ4329I, which was retired in V1R10.

Example

EZD0040I INTERFACE OSAQDIO2 HAS TAKEN OVER ARP RESPONSIBILITY FOR INACTIVE INTERFACE OSAQDIO1

EZD0041I INTERFACE takeback_interface_name HAS TAKEN BACK ARP RESPONSIBILITY FROM INTERFACE takeover_interface_name

Explanation
An interface that previously had its ARP responsibility taken over by another interface became active again, so it took back the ARP responsibility. See the information about interface-layer fault-tolerance for local area networks (interface-takeover function) in z/OS Communications Server: IP Configuration Guide for more information about the interface-takeover function.

In the message text:

takeback_interface_name
The name of the interface that took back the ARP responsibility.

takeover_interface_name
The name of the interface that released the ARP responsibility.

System action
TCP/IP reassigns ARP responsibility to the takeback interface. TCP/IP sends a gratuitous ARP for the IP address of the takeback interface that will reply to ARP requests as the owner.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBIFSTC

Routing code
2,8

Descriptor code
12
Example

EZD0041I INTERFACE OSAQDIO1 HAS TAKEN BACK ARP RESPONSIBILITY FROM INTERFACE OSAQDIO2

EZD0042I AUTOCONFIGURATION OF TEMPORARY ADDRESSES IS DISABLED FOR INTERFACE
interface_name

Explanation

The TCP/IP stack attempted three times to generate a temporary autoconfigured address and to verify that it was unique. During the verification of each of the generated temporary addresses, another node on the link indicated that it was already using the address. Autoconfiguration of temporary addresses is disabled.

In the message text:

interface_name

The name of the IPv6 interface for which autoconfiguration is disabled.

System action

TCP/IP continues.

Operator response

If temporary addresses are not necessary for this interface, no action is required. The interface remains active and autoconfigured public addresses are unaffected. If you want temporary addresses to be generated for the interface, contact the system programmer.

System programmer response

Use the INTERFACE ADDTEMPPREFIX configuration statement to enable temporary address autoconfiguration on the interface for a specific set of prefixes or for all prefixes. Issue a VARY TCPIP,OBEYFILE command to update the configuration. See the information about the INTERFACE statement in z/OS Communications Server: IP Configuration Reference for more information.

User response

Not applicable.

Problem determination

For each of the temporary addresses that was rejected, message EZZ9780I is generated.

Source

z/OS Communications Server TCP/IP

Module

Not applicable.

Routing code

10

Descriptor code

12
Automation
This message goes to the console.

Example

EZD0042I AUTOCONFIGURATION OF TEMPORARY ADDRESSES IS DISABLED FOR INTERFACE QDIO6

EZD0043I RANDOM HISTORY VALUE FOR INTERFACE interface_name GENERATED BY function, ICSF status, ICSF RETURN CODE= return_code, ICSF REASON CODE= reason_code

Explanation
When IPv6 temporary address autoconfiguration is supported for an interface, a random interface ID is generated for the interface. A history value is used as part of the algorithm to generate the random interface ID. The first time that an interface is started a random number generator is used to generate the history value.

If the cryptographic hardware is available, the Integrated Cryptographic Service Facility (ICSF) callable service CSNBRNG generates the history value. If the cryptographic hardware is not available or an error occurs using CSNBRNG, a software random number generator generates the history value.

In the message text:

interface_name
The name of the IPv6 interface for which the history value was generated.

function
Indicates whether the history value was generated by the ICSF callable service CSNBRNG random number generator or by a software random number generator.

status
Indicates whether ICSF is active.

return_code
The return code value, in hexadecimal format, that was returned by the ICSF service. This field is relevant only if the status value is ACTIVE. Otherwise, the value N/A is displayed.

reason_code
The reason code value, in hexadecimal format, that was returned by the ICSF service. This field is relevant only if the status value is ACTIVE. Otherwise, the value N/A is displayed.

System action
TCP/IP continues.

Operator response
Not applicable.

System programmer response
This is an informational message. No action is required.

If the history value was generated by a software random number generator and you want the value to be generated by the ICSF CSNBRNG random number generator, then use the ICSF status, return code, and reason code to determine the problem. When the problem is resolved, the ICSF CSNBRNG random number generator will be used to generate the next history value. The history value is generated only the first time that an interface is started after the TCP/IP stack is started.

User response
Not applicable.
Problem determination

If the status value indicates that ICSF is INACTIVE, start ICSF. If the ICSF return code or reason code is a nonzero value, then see the ICSF return and reason codes information in z/OS Cryptographic Services ICSF Application Programmer's Guide for the specific actions to be taken.

Source

z/OS Communications Server TCP/IP

Module

Not applicable.

Routing code

10

Descriptor code

12

Automation

This message goes to the console.

Example

In this example, ICSF is not active. A software random number generator generated the history value.

```
EZD0043I RANDOM HISTORY VALUE FOR INTERFACE QDIO6 GENERATED BY SOFTWARE , ICSF INACTIVE ,
ICSF RETURN CODE= N/A , ICSF REASON CODE= N/A
```

In this example, ICSF is active and the ICSF module CSNBRNG successfully generated the history value.

```
EZD0043I RANDOM HISTORY VALUE FOR INTERFACE QDIO6 GENERATED BY CSNBRNG , ICSF ACTIVE ,
ICSF RETURN CODE= 0 ICSF REASON CODE= 0
```

In this example, ICSF is active and an unsuccessful attempt was made to generate the history value using the ICSF module CSNBRNG. A software random number generator generated the history value.

```
EZD0043I RANDOM HISTORY VALUE FOR INTERFACE QDIO6 GENERATED BY SOFTWARE , ICSF ACTIVE ,
ICSF RETURN CODE= C , ICSF REASON CODE= 2B34
```

```
EZD0044I INTERFACE interface_name NOT ALLOWED - reason
```

Explanation

The stack could not define the interface because the interface did not adhere to the rules for defining multiple VLAN interfaces to the same OSA-Express port or HiperSockets CHPID. See the information about OSA VLAN or HiperSockets and VLAN in z/OS Communications Server: IP Configuration Guide for more information about these rules. This message provides more details for message EZZ0618I.

In the message text:

interface_name

The name of the interface.

reason

The reason for the rejection. Possible values are:

**NO VLAN ID**

The INTERFACE statement did not specify the VLANID parameter.
NO VLAN ID FOR AN EARLIER DEFINITION
A previous INTERFACE statement for the same OSA-Express port did not specify the VLANID parameter.

VLAN ID NOT UNIQUE
The VLANID parameter on the INTERFACE statement conflicts with the VLANID parameter on a previous INTERFACE statement for the same OSA-Express port.

NOT VMAC ROUTEALL
The INTERFACE statement did not specify the VMAC parameter value ROUTEALL.

NOT VMAC ROUTEALL FOR AN EARLIER DEFINITION
A previous INTERFACE statement for the same OSA-Express port did not specify the VMAC parameter value ROUTEALL.

NO SUBNET MASK
The INTERFACE statement did not specify a subnet mask.

NO SUBNET MASK FOR AN EARLIER DEFINITION
A previous INTERFACE statement for the same OSA-Express port did not specify a subnet mask.

SUBNET NOT UNIQUE
The subnet specified by the INTERFACE overlaps with the subnet specified by a previous INTERFACE statement for the same OSA-Express port.

EXCEEDED MAXIMUM NUMBER OF VLANS
There are already 32 INTERFACE statements for the same OSA-Express port for this IP version.

System action
TCP/IP continues with profile processing, but ignores this INTERFACE statement.

Operator response
Contact the system programmer.

System programmer response
Correct the TCPIP profile to adhere to the rules for defining multiple VLAN interfaces to the same OSA-Express port and restart the stack.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Configuration & Initialization

Module
EZBIFIOC

Routing code
2, 8

Descriptor code
12
EZD0044I INTERFACE OSAQDI02 NOT ALLOWED - VLAN ID NOT UNIQUE

EZD0045I INTERFACE interface_name DOES NOT SUPPORT OLM

Explanation
The OLM parameter was specified on the INTERFACE statement for this device but the OSA-Express adapter microcode level does not support optimized latency mode.

In the message text:

interface_name
   The name of the interface.

System action
TCP/IP activates the device without optimized latency mode enabled.

Operator response
Contact the system programmer.

System programmer response
Either install an OSA-Express Licensed Internal Code (LIC) level that supports optimized latency mode or remove the OLM parameter from the INTERFACE statement. Optimized latency mode is limited to OSA-Express3 ethernet features in QDIO mode running on an IBM System z10. Use the VTAM DISPLAY TRL command to determine the OSA-Express LIC level that is currently installed. See the information about the DISPLAY TRL command in z/OS Communications Server: SNA Operation for more information. See the 2097DEVICE Preventive Service Planning (PSP) bucket and the 2098DEVICE Preventive Service Planning (PSP) bucket for the OSA-Express LIC levels required for optimized latency mode support.

See the information about the optimized latency mode in z/OS Communications Server: IP Configuration Guide for information about the networking affects of configuring with and without optimized latency mode.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBIFIND

Routing code
2,8

Descriptor code
12
**Automation**
None.

**Example**

```
EZD0045I INTERFACE OSAQDI04 DOES NOT SUPPORT OLM
```

**EZD0046I**  ERROR `error_code` ENABLING OLM FOR INTERFACE `interface_name`

**Explanation**

An error occurred while enabling optimized latency mode for the interface.

In the message text:

*error_code*
- The error code returned from the OSA-Express microcode.

*interface_name*
- The name of the interface.

**System action**

TCP/IP activates the device without optimized latency mode enabled.

**Operator response**

Contact the system programmer.

**System programmer response**


**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZBIFIND

**Routing code**

2,8

**Descriptor code**

12

**Automation**

None
Example

EZD0046I  ERROR 0001 ENABLING OLM FOR INTERFACE OSAQDI4

EZD0047I  ERROR error_code REGISTERING DVIPA ipaddr FOR SYSPLEX DISTRIBUTOR WORKLOAD QUEUEING ON INTERFACE interface_name

Explanation
An error occurred while the stack was registering an IP address for sysplex distributor inbound workload queueing.

In the message text:

error_code
The OSA reject code that is returned from the OSA-Express microcode.

ipaddr
The distributed dynamic VIPA that was configured on a VIPADISTRIBUTE statement.

interface_name
The name of the OSA-Express QDIO interface that supports inbound workload queueing.

System action
Sysplex distributor traffic that is received over this interface for this distributed DVIPA is not eligible for inbound workload queueing. Instead, OSA-Express delivers this traffic to the stack using the primary input queue.

Operator response
Contact the system programmer.

System programmer response

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBIFIND

Routing code
2.8

Descriptor code
12
Automation
None.

Example

```
ERROR 0662 REGISTERING DVIPA 197.20.1.1 FOR SYSPLEX DISTRIBUTOR WORKLOAD QUEUEING ON INTERFACE OSAQDIO2

EZD0048I  INTERFACE interface_name DOES NOT SUPPORT QDIO INBOUND WORKLOAD QUEUEING

Explanation
The WORKLOADQ parameter was specified on the INTERFACE statement for this device in the TCP/IP profile. The OSA-Express adapter microcode level for this device does not support QDIO inbound workload queueing.

In the message text:

interface_name
   The name of the interface.

System action
TCP/IP activates the device without QDIO inbound workload queueing enabled.

Operator response
Contact the system programmer.

System programmer response
Either install an OSA-Express Licensed Internal Code (LIC) level that supports QDIO inbound workload queueing or remove the WORKLOADQ parameter from the INTERFACE statement. QDIO inbound workload queueing is limited to OSA-Express3 ethernet features in QDIO mode (CHPID type OSD) that are running on an IBM System z10. Use the VTAM DISPLAY TRL command to determine what OSA-Express LIC level is currently installed. See the information about the DISPLAY TRL command in z/OS Communications Server: SNA Operation for more information. See the 2097DEVICE Preventive Service Planning (PSP) bucket and the 2098DEVICE Preventive Service Planning (PSP) bucket for the OSA-Express LIC levels required for QDIO inbound workload queueing support.

See the information about the QDIO inbound workload queueing in z/OS Communications Server: IP Configuration Guide for information about the networking affects of configuring with and without the WORKLOADQ parameter.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBIFIND
Routing code
2,8

Descriptor code
12

Automation
None.

Example
EZD0048I INTERFACE OSAQDIO4 DOES NOT SUPPORT QDIO INBOUND WORKLOAD QUEUEING

EZD0049I ERROR error_code registers ipaddr and port port for EE WORKLOAD QUEUEING ON INTERFACE interface_name

Explanation
An error occurred while the stack was registering an IP address and port for Enterprise Extender inbound workload queueing.

In the message text:

error_code
The OSA reject code returned from the OSA-Express microcode.

ipaddr
The static VIPA used for Enterprise Extender.

port
The UDP port used for Enterprise Extender.

interface_name
The name of the OSA-Express QDIO interface that supports inbound workload queueing.

System action
Enterprise Extender traffic that is received over this interface for this IP address and port is not eligible for inbound workload queueing. Instead, OSA-Express delivers this traffic to the stack by using the primary input queue.

Operator response
Contact the system programmer.

System programmer response

User response
Not applicable.

Problem determination
Not Applicable.
Source
z/OS Communications Server TCP/IP

Module
EZBIFIND

Routing code
2, 8

Descriptor code
12

Automation
Not applicable.

Example
ERROR 0002 REGISTERING IP ADDR 200.1.1.1 AND PORT 12000 FOR EE WORKLOAD QUEUEING ON INTERFACE OSAQIO2

EZD0101I    NETSTAT versionRelease

Explanation
This message displays the current version and release for the command being displayed in the LONG format. The message is followed by the output for the requested command report. See the Netstat report details and examples in z/OS Communications Server: IP System Administrator's Commands for a detailed description of the report.

System action
The Display Netstat command continues.

Operator response
None.

System programmer response
None.

Module
EZACDNE6

Procedure name
procACCN6(), procALLC6(), procARP(), procBYTE6(), procCACH6(), procCNFG6(), procCONN6(), procDEVL6(), procDVCF6(), procHOME6(), procIDS6(), procND6(), procPORT6(), procROUTE6(), procSOCK6(), procSTAT6(), procVCRT6(), procVDPT6(), procVIPA6()

EZD0800I    proc_name IP ADDRESS ip_address IS NOT VALID FOR USE IN ENTERPRISE EXTENDER

Explanation
The IP address is not valid. The IP address was not defined to the TCP/IP stack or was defined as a loopback address.
proc_name is the name of the started task associated with the TCP/IP address space.

ip_address is the Internet protocol address defined in SNA for the Enterprise Extender connection.

**System action**
The Enterprise Extender connection using this IP address as the local address fails.

**Operator response**
None.

**System programmer response**
Ensure that the IP address for Enterprise Extender defined in SNA matches a static VIPA address defined in this TCP/IP stack. The IP address is defined in SNA using the IPADDR or HOSTNAME start options or the IPADDR or HOSTNAME GROUP operands in the XCA major node. For more information, see z/OS Communications Server: SNA Resource Definition Reference.

**Module**
EZBUDBYP

**Procedure name**
OpenReq

EVD0801I  proc_name IP ADDRESS ip_address IS NOT A VALID STATIC VIPA ADDRESS

**Explanation**
The local IP address requested for an Enterprise Extender connection is not defined as one of the static VIPA addresses in this TCP/IP stack.

proc_name is the name of the started task associated with the TCP/IP address

space. ip_address is the Internet protocol address defined in SNA for the Enterprise Extender connection.

**System action**
The Enterprise Extender connection using this IP address as the local address fails.

**Operator response**
None.

**System programmer response**
Ensure that the Enterprise Extender definitions defined in SNA match the static VIPA definitions in your TCP/IP profile. The IP address is defined in SNA using the IPADDR or HOSTNAME start options or the IPADDR or HOSTNAME GROUP operands in the XCA major node. For more information, see z/OS Communications Server: SNA Resource Definition Reference.

**Module**
EZBUDBYP

**Procedure name**
OpenReq

EVD0802I  proc_name ENTERPRISE EXTENDER CONNECTION FAILURE ON IP ADDRESS ip_address

**REASON reason**
Explanation
An Enterprise Extender connection using this IP address failed for the reason indicated.

In the message text:

**proc_name**
The name of the started task associated with the TCP/IP address space.

**ip_address**
The IP address defined in SNA for the Enterprise Extender connection.

**reason**
A code identifying the reason for the failure. Possible values are:

- **X'08'**
  The IP address requested or used by the caller is not a valid IP address for the stack.

- **X'0C'**
  The IP address requested or used by the caller is not a valid VIPA address for the stack.

- **X'10'**
  The port number requested by the caller cannot be reserved.

- **X'14'**
  The parameter passed by the caller is not valid for the requested operation.

- **X'18'**
  The state of the UDP connection is not valid for the requested operation.

- **X'1C'**
  The requested destination for the datagram is unreachable.

- **X'20'**
  The caller is not authorized to perform the requested operation.

- **X'24'**
  The requested operation cannot be completed as a result of a storage failure.

- **X'28'**
  The requested datagram to be sent exceeds the stack's maximum, or a received datagram has a data length of zero.

- **X'40'**
  The requested operation could not be completed because of a permanent error.

**System action**
The Enterprise Extender connection using this IP address as the local address fails.

**System programmer response**
Save the system log and contact IBM service.

**User response**
Contact the system programmer.

**Module**
EZBIFIUM

**Procedure name**
ClearMPC

**EZD0810I**  IPSec Suppressed logging of _number_ packet message(s) due to buffer overflow: _timestamp_
Explanation
The stack IPSec log buffer overflowed. Log entries for the number of packets indicated cannot be logged. 

*number* is the number of log entries.

*timestamp* indicates when the buffer overflow occurred. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

System action
TCP/IP processing continues.

Operator response
Contact the system programmer.

System programmer response
You might be logging messages that are not required or your IP Security stack might be under a denial-of-service attack. Ensure that you are logging only items of interest. For example, broadcast messages should have a deny rule with the `IpFilterLogging` option set to `no` to prevent filling up the log. If you are under attack, log processing will return to normal when the attack ends.

Module
EZATRMD

Procedure name
trmd_ipsec_log

**EZD0811I** Decapsulation failed: *timestamp* sipaddr= *sipaddr* dipaddr= *dipaddr* proto= *proto* vpnaction= *vpnaction* tunelID= *tunID* AHIndex= *AHIndex* ESPSPI= *ESPIndex* rsn= *rsn* ICSF Return Code= *return_code* ICSF Reason Code = *reason_code* ikeport= *ikeport*

Explanation
The IPSec packet cannot be decapsulated by the receiving stack and is discarded.

*timestamp* indicates when the decapsulation failure occurred. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

*sipaddr* is the source IP address.

*dipaddr* is the destination IP address.

*proto* is the protocol. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
- AH(51)
- ICMPv6(58)
- OSPF(89)
• IPIP(94)
  • The protocol number

`vpnaction` is the `vpnaction` name. If no tunnel is found, `vpnaction` displays N/A.

• In the Policy Agent configuration file:
  – If the tunnel is a manual tunnel, `vpnaction` is the name specified on the `IpManVpnAction` statement.
  – If the tunnel is a dynamic tunnel, `vpnaction` is the name specified on the `IpDynVpnAction` statement.

• When configured with the IBM Configuration Assistant for z/OS Communications Server, the `vpnaction` name corresponds to the name of the security level in the GUI. The `vpnaction` name also contains a suffix appended to the security level name to guarantee uniqueness.

`tunID` is the tunnel ID. If the value of `vpnaction` is N/A, a tunnel with matching end points and security parameter indices (spi) could not be found.

`AHindex` is the AH security parameter index.

`ESPindex` is the ESP security parameter index.

`rsn` indicates the specific reason decapsulation failed.

<table>
<thead>
<tr>
<th><code>rsn</code> Value</th>
<th>Explanation</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Decryption failed</td>
<td>This problem might be caused by a transmission error or by a sender error. For manual tunnels, this might be the result of a policy definition error.</td>
</tr>
<tr>
<td>2</td>
<td>AH authentication failed</td>
<td>This problem might be caused by a transmission error or by a sender error. For manual tunnels, this might be the result of a policy definition error. This problem might also be caused by a failure in an ICSF service. If so, the specific failure will be reported on the ICSF Return Code and ICSF Reason Code fields.</td>
</tr>
<tr>
<td>3</td>
<td>ESP authentication failed</td>
<td>This problem might be caused by a transmission error or by a sender error. For manual tunnels, this might be the result of a policy definition error. This problem might also be caused by a failure in an ICSF service. If so, the specific failure will be reported on the ICSF Return Code and ICSF Reason Code fields.</td>
</tr>
<tr>
<td>4</td>
<td>Out of Replay window</td>
<td>A transmission error might have occurred or a packet might have been delayed.</td>
</tr>
<tr>
<td>6</td>
<td>Unknown authentication algorithm</td>
<td>An internal error occurred while authenticating the packet.</td>
</tr>
<tr>
<td>7</td>
<td>Unknown encryption algorithm</td>
<td>An internal error occurred while decrypting the packet.</td>
</tr>
<tr>
<td>8</td>
<td>No tunnel found for AH SPI</td>
<td>This message might be the result of a timing condition. On tunnel activation this message might be seen if packets are sent while one tunnel endpoint has the tunnel installed and the other tunnel endpoint does not. In this case, this is a transient condition and no action is required. For manual tunnels, this might be the result of a policy definition error. This message can also be the result of a transmission error or a sender error.</td>
</tr>
<tr>
<td><strong>rsn</strong> Value</td>
<td><strong>Explanation</strong></td>
<td><strong>Comments</strong></td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>9</td>
<td>No tunnel found for ESP SPI</td>
<td>This message might be the result of a timing condition. On tunnel activation this message might be seen if packets are sent while one tunnel endpoint has the tunnel installed and the other tunnel endpoint does not. In this case, this is a transient condition and no action is required. For manual tunnels, this might be the result of a policy definition error. This message can also be the result of a transmission error or a sender error.</td>
</tr>
</tbody>
</table>
| 10           | More than one tunnel matched during decapsulation                           | This might be due to one of the following problems:  
• For manual tunnels, there is an error in the spi values specified in the policy definition.  
• The packet is protected by nested tunnels and z/OS is the endpoint for more than 1 of these tunnels. This configuration is not supported by z/OS. |
| 11           | IPSec headers did not match tunnel definition                                 | The policy, either the AH policy, the ESP policy, or both, defined for the tunnel did not match the IPSec protocols in the packet. For example, the policy specified encryption was required but no ESP header was found. For manual tunnels, this is most likely a policy definition error. |
| 12           | AH and ESP headers not in expected sequence                                  | For manual tunnels, this might be a result of a policy definition error.                                                                                                                                 |
| 13           | Storage shortage                                                              | Storage to complete the request is not currently available. Until the storage shortage is relieved, decapsulation will fail.                                                                               |
| 14           | Encrypted data length is not a multiple of 8 bytes or 16 bytes if AES encryption or decryption is being used | This problem might be caused by a transmission error or by faulty encryption of the data by the sender.                                                                                                     |
| 15           | No data was sent in the packet                                                | This problem might be caused by a transmission error or by a sender error.                                                                                                                                 |
| 16           | The packet is too small to contain the AH or ESP header                       | This problem might be caused by a transmission error or by a sender error.                                                                                                                                 |
| 17           | Invalid IP option length                                                      | This problem might be caused by a transmission error or by a sender error.                                                                                                                                 |
| 18           | UDP Encapsulation mismatch                                                    | Either the packet was UDP-encapsulated and the tunnel did not indicate UDP encapsulation or the packet was not UDP-encapsulated and the tunnel expected UDP encapsulation.                                           |
| 19           | Nested UDP-encapsulated headers                                               | This configuration is not supported by z/OS.                                                                                                                                                               |
| 24           | Failure in ICSF Service                                                       | The ICSF Return Code and the ICSF Reason Code fields contain the return and reason codes that were returned from the ICSF service.                                                                       |
| 25           | ICSF is not available                                                         | Either ICSF is not active or it has not completed initialization.                                                                                                                                           |
| 26           | Encapsulation mismatch                                                       | The packet encapsulation did not match the local tunnel policy.                                                                                                                                              |
**return_code** is the return code value, in hexadecimal format, returned from the ICSF service.

**reason_code** is the reason code value, in hexadecimal format, returned from the ICSF service.

**ikeport** is the source port from the UDP encapsulation header. If the packet is not UDP encapsulated, the ikeport value is N/A.

**System action**
TCP/IP processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
The system programmer response depends on the rsn value:

<table>
<thead>
<tr>
<th>rsn Value</th>
<th>System programmer response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 2, 3</td>
<td>If the problem is transient, no action is required. For manual tunnels, verify that the security parameters and encryption keys on the IpManVpnAction statement are correctly defined. When configured with the IBM Configuration Assistant for z/OS Communications Server, the IpManVpnAction name corresponds to the name of the security level in the GUI. The IpManVpnAction name also contains a suffix that is appended to the security level name to guarantee uniqueness. For more details about the IpManVpnAction statement, see the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Otherwise, ensure that the tunnel is defined correctly on the sending and receiving systems. Use the <code>ipsec</code> command to display filter and tunnel information. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the <code>man ipsec</code> command in a z/OS UNIX shell to obtain information about the <code>ipsec</code> command syntax and options. If the problem was due to an ICSF failure, then see the information about the ICSF and TSS Return and Reason Codes in z/OS Cryptographic Services ICSF Application Programmer's Guide for the specific actions to be taken.</td>
</tr>
<tr>
<td>4</td>
<td>If the problem is transient, no action is required. Otherwise, ensure that the tunnel is defined and activated correctly on the sending and receiving systems. Use the <code>ipsec</code> command to display filter and tunnel information. The <code>ipsec</code> command can also be used to refresh or activate a tunnel. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the <code>man ipsec</code> command in a z/OS UNIX shell to obtain information about the <code>ipsec</code> command syntax and options.</td>
</tr>
<tr>
<td>6, 7</td>
<td>Contact the IBM Software Support Center.</td>
</tr>
<tr>
<td>8, 9</td>
<td>If the problem is transient, no action is required. For manual tunnels, verify the security parameters and encryption keys on the IpManVpnAction statement are correctly defined. For more details about the IpManVpnAction statement, see the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Otherwise, ensure that the tunnel is defined and activated correctly on the sending and receiving systems. Use the <code>ipsec</code> command to display filter and tunnel information. The <code>ipsec</code> command can also be used to refresh or activate a tunnel. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the <code>man ipsec</code> in a z/OS UNIX shell to obtain information about the <code>ipsec</code> command syntax and options.</td>
</tr>
<tr>
<td>rsn Value</td>
<td>System programmer response</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>10</td>
<td>Ensure that the tunnel is defined correctly on the sending and receiving systems. Use the <code>ipsec</code> command to display filter and tunnel information. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the <code>man ipsec</code> command in a z/OS UNIX shell to obtain information about the <code>ipsec</code> command syntax and options.</td>
</tr>
<tr>
<td>11, 12</td>
<td>For manual tunnels, verify the security parameters and encryption keys on the IpManVpnAction statement are correctly defined. For more details about the IpManVpnAction statement, see the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Otherwise, ensure that the tunnel is defined correctly on the sending and receiving systems. Use the <code>ipsec</code> command to display filter and tunnel information. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the <code>man ipsec</code> command in a z/OS UNIX shell to obtain information about the <code>ipsec</code> command syntax and options.</td>
</tr>
<tr>
<td>13</td>
<td>Determine the cause of the storage shortage.</td>
</tr>
<tr>
<td>14, 15, 16, 17</td>
<td>If the problem is transient, no action is required. Otherwise, ensure that the tunnel is defined correctly on the sending and receiving systems. Use the <code>ipsec</code> command to display filter and tunnel information. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the <code>man ipsec</code> command in a z/OS UNIX shell to obtain information about the <code>ipsec</code> command syntax and options.</td>
</tr>
<tr>
<td>18, 19</td>
<td>Ensure that the tunnel is defined correctly on the sending and receiving systems. Use the <code>ipsec</code> command to display filter and tunnel information. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the <code>man ipsec</code> command in a z/OS UNIX shell to obtain information about the <code>ipsec</code> command syntax and options.</td>
</tr>
<tr>
<td>24</td>
<td>See the information about the ICSF and TSS Return and Reason Codes in z/OS Cryptographic Services ICSF Application Programmer's Guide for the specific actions to be taken.</td>
</tr>
<tr>
<td>25</td>
<td>Start ICSF if it is not active.</td>
</tr>
<tr>
<td>26</td>
<td>Contact the IBM Software Support Center.</td>
</tr>
</tbody>
</table>

**Module**

EZATRMD

**Procedure name**

trmd_ipsec_log

**EZD0812I** Tunnel deleted: timestamp vpnaction= vpnaction tunnelID= tunID AHSPI= AH_index ESPSPI= ESPIndex

**Explanation**

The specified tunnel is deleted from the stack.

`timestamp` indicates when the tunnel was deleted. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

`vpnaction` is the vpnaction name.

- In the policy agent configuration file, the `vpnaction` value is one of the following:
- If the tunnel is a manual tunnel, `vpnaction` is the name specified on the `IpManVpnAction` statement.
- If the tunnel is a dynamic tunnel, `vpnaction` is the name specified on the `IpDynVpnAction` statement.

- When configured with the IBM Configuration Assistant for z/OS Communications Server, the `vpnaction` name corresponds to the name of the security level in the GUI. The `vpnaction` name also contains a suffix appended to the security level name to guarantee uniqueness.

`tunID` is the tunnel ID.

`AH_index` is the AH security parameter index.

`ESPindex` is the ESP security parameter index.

**System action**

TCP/IP processing continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

EZATRMD

**Procedure name**

trmd_ipsec_log

**Explanation**

The specified tunnel expired. The tunnel was marked inactive but was not deleted from the stack.

`timestamp` indicates the time the tunnel expired. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

`vpnaction` is the vpnaction name.

- In the policy agent configuration file:
  - If the tunnel is a manual tunnel, `vpnaction` is the name specified on the `IpManVpnAction` statement.
  - If the tunnel is a dynamic tunnel, `vpnaction` is the name specified on the `IpDynVpnAction` statement.

- When configured with the IBM Configuration Assistant for z/OS Communications Server, the `vpnaction` name corresponds to the name of the security level in the GUI. The `vpnaction` name also contains a suffix appended to the security level name to guarantee uniqueness.

`tunID` is the tunnel ID.

`AH_index` is the AH security parameter index.

`ESPindex` is the ESP security parameter index.

**System action**

TCP/IP processing continues.
Operator response
None.

System programmer response
None.

Module
EZATRMD

Procedure name
trmd_ipsec_log

**EZD0814I** Packet permitted: timestamp filter rule= rulename ext= instance sipaddr= sipaddr dipaddr= dipaddr proto= proto tag1 tag2 tag3 Interface= ifaddr ( dir ) secclass= secclass dest= dest len= len vpnaction= vpnaction tunnelID= tunID ifcname= ifcname fragment= frag

Explanation
An IP packet matched the indicated filter rule and was permitted. For this message to be written, the matched filter rule must have IpFilterLogging set to yes.

*timestamp* is the stack timestamp that indicates the time at which the IP packet was processed by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

*rulename* is the filter rule name. If the IP packet matched a dynamic filter rule, the rule name of the corresponding anchor filter rule will be displayed; otherwise, the rule name of the matching filter rule will be displayed.

- In the policy agent configuration file, *rulename* is the name specified on the IpFilterRule statement.
- When configured with the IBM Configuration Assistant for z/OS Communications Server, *rulename* corresponds to the name of a Connectivity Rule in the GUI. *rulename* also contains a suffix appended to the Connectivity Rule name to guarantee uniqueness.

*instance* is the rule name extension that indicates which instance of the rule name was matched.

*sipaddr* is the source IP address.

*dipaddr* is the destination IP address.

*proto* is the protocol of the packet. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
- AH(51)
- ICMPv6(58)
- OSPF(89)
- IPIP(94)
- MIPv6(135)
- Unknown
- The protocol number
The tag1 value varies depending on the proto value.

- If the proto value is ICMP or ICMPv6, the tag1 value is type= followed by the ICMP or ICMPv6 type, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the proto value is TCP or UDP, the tag1 value is sport= followed by the source port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
- If the proto value is OSPF, the tag1 value is type= followed by the type, or followed by the value Unknown if the OSPF header is not present in the packet as the result of fragmentation.
- If the proto value is MIPv6, the tag1 value is type= followed by the type, or followed by the value Unknown if the MIPv6 header is not present in the packet as the result of fragmentation.
- If the proto value is any value not previously mentioned, the tag1 value is -= which indicates that the data is not applicable.

tag2 value varies depending on the proto value.

- If the proto value is ICMP or ICMPv6, the tag2 value is code= followed by the ICMP or ICMPv6 code, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the proto value is TCP or UDP, the tag2 value is dport= followed by the destination port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
- If the proto value is any value not previously mentioned, the tag2 value is -= which indicates that the data is not applicable.

tag3 value varies depending on the proto value and direction.

- If the proto value is TCP or UDP, the direction is inbound, and the port has been translated by the CommServer NAT Traversal function, the tag3 value is origport= followed by the original source port.
- If the proto value is TCP or UDP, the direction is outbound, and the port has been translated by the CommServer NAT Traversal function, the tag3 value is origport= followed by the original destination port.
- If the proto value is any value not previously mentioned, the tag3 value is -= which indicates that the data is not applicable.

ifcaddr is the interface address over which the packet was received or sent.

dir is I if packet is inbound, O if packet is outbound.

secclass is the security class assigned to the interface. Security class is a numeric value in the range of 0–255.

dest is local if a local destination or routed if being routed.

len is the packet length.

vpnaction is the vpnaction name. If no tunnel is associated with the matched filter, vpnaction displays N/A.

- In the policy agent configuration file, the vpnaction value is one of the following:
  - If the tunnel is a manual tunnel, vpnaction is the name specified on the IpManVpnAction statement.
  - If the tunnel is a dynamic tunnel, vpnaction is the name specified on the IpDynVpnAction statement.
- When configured with the IBM Configuration Assistant for z/OS Communications Server, the vpnaction name corresponds to the name of the security level in the GUI. The vpnaction name also contains a suffix appended to the security level name to guarantee uniqueness.

tunID is the tunnel ID.

ifcname is the interface name.

frag specifies whether the packet is a fragment. The value is Y if the packet is a fragment, or N if the packet is not a fragment.

System action

TCP/IP processing continues.
Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: TRMD

Module
EZATRMD

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

```
EZD0814I Packet permitted: 07/05/2007 16:19:44.39 filter rule= ipsec-2 ext= 1 sipaddr= 9.42.130.185
dipaddr= 10.1.1.1 proto= tcp(6) sport= 80 dport= 1026 - Interface= 9.1.1.1 (O) secclass= 255
dest= local len= 284 vpnaction= DynAction tunnelID= Y4 ifcname= TRLE1AL fragment= N
```

```
EZD0815I Packet denied by policy: timestamp filter rule= rulename ext= instance sipaddr= sipaddr
dipaddr= dipaddr proto= proto tag1 tag2 tag3 Interface= ifcaddr ( dir ) secclass= secclass
dest= dest len= len vpnaction= vpnaction tunnelID= tunID ifcname= ifcname fragment= frag
```

Explanation
An IP packet matched the indicated deny filter rule. For this message to be written, the matched filter rule must have IpFilterLogging set to yes.

timestamp is the stack timestamp that indicates the time at which the IP packet was processed by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

rulename is the filter rule name. If the IP packet matched a dynamic filter rule, the rule name of the corresponding anchor filter rule will be displayed; otherwise, the rule name of the matching filter rule will be displayed.

• In the policy agent configuration file, rulename is the name specified on the IpFilterRule statement.
When configured with the IBM Configuration Assistant for z/OS Communications Server, *rulename* corresponds to the name of a Connectivity Rule in the GUI. *rulename* also contains a suffix appended to the Connectivity Rule name to guarantee uniqueness.

*instance* is the rule name extension that indicates which instance of the rule name was matched.

*sipaddr* is the source IP address.

*dipaddr* is the destination IP address.

*proto* is the protocol of the packet. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
- AH(51)
- ICMPv6(58)
- OSPF(89)
- MIPv6(135)
- IPIP(94)
- Unknown

The *tag1* value varies depending on the *proto* value.

- If the *proto* value is ICMP or ICMPv6, the *tag1* value is *type* = followed by the ICMP or ICMPv6 type, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the *proto* value is TCP or UDP, the *tag1* value is *sport* = followed by the source port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
- If the *proto* value is OSPF, the *tag1* value is *type* = followed by the type, or followed by the value Unknown if the OSPF header is not present in the packet as the result of fragmentation.
- If the *proto* value is MIPv6, the *tag1* value is *type* = followed by the type, or followed by the value Unknown if the MIPv6 header is not present in the packet as the result of fragmentation.
- If the *proto* value is any value not previously mentioned, the *tag1* value is -= which indicates that the data is not applicable.

The *tag2* value varies depending on the *proto* value.

- If the *proto* value is ICMP or ICMPv6, the *tag2* value is *code* = followed by the ICMP or ICMPv6 code, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the *proto* value is TCP or UDP, the *tag2* value is *dport* = followed by the destination port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
- If the *proto* value is any value not previously mentioned, the *tag2* value is -= which indicates that the data is not applicable.

The *tag3* value varies depending on the *proto* value and direction.

- If the *proto* value is TCP or UDP, the direction is inbound, and the port has been translated by the CommServer NAT Traversal function, the *tag3* value is *origsport* = followed by the original source port.
- If the *proto* value is TCP or UDP, the direction is outbound, and the port has been translated by the CommServer NAT Traversal function, the *tag3* value is *origsport* = followed by the original destination port.
- If the *proto* value is any value not previously mentioned, the *tag3* value is -= which indicates that the data is not applicable.
iffaddr is the interface address over which the packet was received or sent.

dir is I if packet is inbound, O if packet is outbound.

secclass is the security class assigned to the interface. Security class is a numeric value in the range of 0–255.

dest is local if a local destination or routed if being routed.

len is the packet length.

vpnaction is the vpnaction name. If no tunnel is associated with the matched filter, vpnaction displays N/A.

- In the policy agent configuration file, the vpnaction value is one of the following:
  - If the tunnel is a manual tunnel, vpnaction is the name specified on the IpManVpnAction statement.
  - If the tunnel is a dynamic tunnel, vpnaction is the name specified on the IpDynVpnAction statement.

- When configured with the IBM Configuration Assistant for z/OS Communications Server, the vpnaction name corresponds to the name of the security level in the GUI. The vpnaction name also contains a suffix appended to the security level name to guarantee uniqueness.

ifcname is the interface name

tunID is the tunnel ID.

frag specifies whether the packet is a fragment. The value is Y if the packet is a fragment, or N if the packet is not a fragment.

System action
TCP/IP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: TRMD

Module
EZATRMD

Routing code
Not applicable.

Descriptor code
Not applicable.
**Automation**
Not applicable.

**Example**

EZD0815I Packet denied by policy: 07/05/2007 16:19:44.39 filter rule= deny-2 ext= 1 sipaddr= 9.42.130.185 dipaddr= 10.1.1.1 proto= tcp(6) sport= 1026 dport= 80 - Interface= 9.1.1.1 (I) secclass= 255 dest= local len= 284 vpnaction= N/A tunnelID= N/A ifcname= TRLE1AL fragment= N

EZD0816I IPSec Policy updated: timestamp type= policy_type status= policy_status

**Explanation**

The Pagent policy or Default policy was updated.

*timestamp* indicates the time the policy change occurred in the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

*policy_type* indicates the type of policy updated. Valid types are Pagent or Default.

*policy_status* indicates whether the policy type updated was active or inactive. Possible values are Active or Inactive. For example, if the default policy was updated but the policy that was active at the time of the update was provided by pagent, *policy_status* will be Inactive.

**System action**
TCP/IP processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZATRMD

**Procedure name**
trmd_ipsec_log

EZD0817I IPSec Policy switched to policy_type policy: timestamp

**Explanation**

The type of policy being used was changed as a result of the *ipsec -f default* or *ipsec -f reload* command.

*policy_type* indicates the type of policy that is now active as a result of the policy switch. Possible values are:

**Pagent**
Indicates that the new policy is the IPSEC policy defined in Pagent.

**Default**
Indicates that the new policy is the IPSEC policy defined in the TCP/IP configuration file.
timestamp indicates the time the policy change occurred in the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

**System action**
TCP/IP processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZATRMD

**Procedure name**
trmd_ipsec_log

**EZD0818I** Tunnel added: timestamp vpnaction= vpnaction tunnelID= tunID AHSPI= AHIndex ESPSPI= ESPIndex

**Explanation**
The specified tunnel was added to the stack.

timestamp indicates the time at which the tunnel was added by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

vpnaction is the vpnaction name.

- In the policy agent configuration file, the vpnaction value is one of the following:
  - If the tunnel is a manual tunnel, vpnaction is the name specified on the IpManVpnAction statement.
  - If the tunnel is a dynamic tunnel, vpnaction is the name specified on the IpDynVpnAction statement.
- When configured with the IBM Configuration Assistant for z/OS Communications Server, the vpnaction name corresponds to the name of the security level in the GUI. The vpnaction name also contains a suffix appended to the security level name to guarantee uniqueness.

tunID is the tunnel ID.

AHIndex is the AH security parameter index.

ESPIndex is the ESP security parameter index.

**System action**
TCP/IP processing continues.

**Operator response**
None.

**System programmer response**
None.
Module
EZATRMD

Procedure name
trmd_ipsec_log

EZD0819I  Tunnel activated: timestamp vpnaction= vpnaction tunnelID= tunID AHSPI= AHIndex
           ESPSPI= ESPIndex

Explanation
The specified manual tunnel was activated in the stack.

- timestamp indicates the time at which the tunnel was activated. This time is retrieved from the system's time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.
- vpnaction is the vpnaction name.
  - In the policy agent configuration file, vpnaction is the name specified on the IpManVpnAction statement.
  - When configured with the IBM Configuration Assistant for z/OS Communications Server, the vpnaction name corresponds to the name of the security level in the GUI. The vpnaction name also contains a suffix appended to the security level name to guarantee uniqueness.
- tunID is the tunnel ID.
- AHIndex is the AH security parameter index.
- ESPIndex is the ESP security parameter index.

System action
TCP/IP processing continues.

Operator response
None.

System programmer response
None.

Module
EZATRMD

Procedure name
trmd_ipsec_log

EZD0820I  Tunnel deactivated: timestamp vpnaction= vpnaction tunnelID= tunID AHSPI= AHIndex
           ESPSPI= ESPIndex

Explanation
The specified manual tunnel was deactivated but was not deleted from the stack.

- timestamp indicates the time at which the tunnel was deactivated. This time is retrieved from the system's time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.
- vpnaction is the vpnaction name.
• In the policy agent configuration file, *vpnaction* is the name specified on the IpManVpnAction statement.

• When configured with the IBM Configuration Assistant for z/OS Communications Server, the *vpnaction* name corresponds to the name of the security level in the GUI. The *vpnaction* name also contains a suffix appended to the security level name to guarantee uniqueness.

*vpnaction* is the tunnel ID.

*AHindex* is the AH security parameter index.

*ESPindex* is the ESP security parameter index.

**System action**

TCP/IP processing continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

EZA**TRMD**

**Procedure name**

*trmd_ipsec_log*

**EZD0821I**  Packet denied, no tunnel: *timestamp* filter rule= *rulename* ext= *instance* sipaddr= *sipaddr* dipaddr= *dipaddr* proto= *proto* tag1 tag2 tag3 Interface= *ifcaddr* (dir) secclass= *secclass* dest= *dest* len= *len* vpnaction= *vpnaction* ifcname= *ifcname* fragment= *frag*

**Explanation**

An IP packet matched the indicated filter rule but no matching tunnel was found. For this message to be written, the matched filter rule must have IpFilterLogging set to *yes*.

*timestamp* is the stack timestamp that indicates the time at which the IP packet was processed by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

*rulename* is the filter rule name. If the IP packet matched a dynamic filter rule, the rule name of the corresponding anchor filter rule will be displayed; otherwise, the rule name of the matching filter rule will be displayed.

• In the policy agent configuration file, *rulename* is the name specified on the IpFilterRule statement.

• When configured with the IBM Configuration Assistant for z/OS Communications Server, *rulename* corresponds to the name of a Connectivity Rule in the GUI. *rulename* also contains a suffix appended to the Connectivity Rule name to guarantee uniqueness.

*instance* is the rule name extension that indicates which instance of the rule name was matched.

*sipaddr* is the source IP address.

*dipaddr* is the destination IP address.

*proto* is the protocol from the packet. Possible values are:

• ICMP(1)
• IGMP(2)
• IP(4)
• TCP(6)
• UDP(17)
• ESP(50)
• AH(51)
• ICMPv6(58)
• OSPF(89)
• IPIP(94)
• MIPv6(135)
• Unknown
• The protocol number

The tag1 value varies depending on the proto value.

• If the proto value is ICMP or ICMPv6, the tag1 value is type= followed by the ICMP or ICMPv6 type, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
• If the proto value is TCP or UDP, the tag1 value is sport= followed by the source port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
• If the proto value is OSPF, the tag1 value is type= followed by the type, or followed by the value Unknown if the OSPF header is not present in the packet as the result of fragmentation.
• If the proto value is MIPv6, the tag1 value is type= followed by the type, or followed by the value Unknown if the MIPv6 header is not present in the packet as the result of fragmentation.
• If the proto value is any value not previously mentioned, the tag1 value is -= which indicates that the data is not applicable.

tag2 value varies depending on the proto value.

• If the proto value is ICMP or ICMPv6, the tag2 value is code= followed by the ICMP or ICMPv6 code, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
• If the proto value is TCP or UDP, the tag2 value is dport= followed by the destination port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
• If the proto value is any value not previously mentioned, the tag2 value is -= which indicates that the data is not applicable.

tag3 value varies depending on the proto value and direction.

• If the proto value is TCP or UDP, the direction is inbound, and the port has been translated by the CommServer NAT Traversal function, the tag3 value is origport= followed by the original source port.
• If the proto value is TCP or UDP, the direction is outbound, and the port has been translated by the CommServer NAT Traversal function, the tag3 value is origport= followed by the original destination port.
• If the proto value is any value not previously mentioned, the tag3 value is -= which indicates that the data is not applicable.

ifcaddr is the interface address over which the packet was received or sent.

dir is I if packet is inbound, O if packet is outbound.

secclass is the security class assigned to the interface. Security class is a numeric value in the range of 0–255.
dest is local if a local destination or routed if being routed.

len is the packet length.

vpnaction is the vpnaction name. If no tunnel is associated with the matched filter, vpnaction displays N/A.

• In the policy agent configuration file, the vpnaction value is one of the following:
  – If the tunnel is a manual tunnel, vpnaction is the name specified on the IpManVpnAction statement.
  – If the tunnel is a dynamic tunnel, vpnaction is the name specified on the IpDynVpnAction statement.
When configured with the IBM Configuration Assistant for z/OS Communications Server, the *vpnaction* name corresponds to the name of the security level in the GUI. The *vpnaction* name also contains a suffix appended to the security level name to guarantee uniqueness.

*ifcname* is the interface name

*frag* specifies whether the packet is a fragment. The value is *Y* if the packet is a fragment, or *N* if the packet is not a fragment.

**System action**

TCP/IP processing continues.

**Operator response**

Contact the system programmer.

**System programmer response**

For manual tunnels, verify that the IPSecurity policy defined a tunnel and that the time conditions are correct. For dynamic tunnels, this can message can occur if the tunnel is not found and *AllowOnDemand No* is specified in the policy. If this traffic should be allowed, either activate the tunnel using the *ipsec* command or change the policy to allow OnDemand negotiations of Security Associations.

• In the policy agent configuration file, take the following actions:
  – Set the time conditions by using the *IpTimeCondition* statement. Time conditions can be included in an *IpFilterRule* statement or in an *IpManVpnAction* statement.
  – Set *AllowOnDemand* on either the *IpFilterPolicy* statement or on an *IpLocalStartAction* statement.
• When configured with the IBM Configuration Assistant for z/OS Communications Server, take the following actions:
  – Set the time conditions in the Advanced Settings panel of a security level that is defined as a manual tunnel or in the Connectivity Rule Advanced IPSec: Filter Logging / Effective Time panel
  – Set *AllowOnDemand* on the Connectivity Rule Advanced IPSec: Dynamic Tunnels: How to Activate panel.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: TRMD

**Module**

EZATRZOS

**Routing code**


**Descriptor code**


**Automation**

Not applicable.
Example

EZD0821I Packet denied, no tunnel: 07/05/2007 16:19:44.39 filter rule= ipsec-2 ext= 1 sipaddr= 9.42.130.185 dipaddr= 10.1.1.1 proto= tcp(6) sport= 80 dport= 1026 -= Interface= 9.1.1.1 (O) secclass= 255 dest= local len= 284 vpnaction= DynAction ifcname= TRLE1AL fragment= N

EZD0822I Packet denied, tunnel inactive: timestamp filter rule= rulename ext= instance sipaddr= sipaddr dipaddr= dipaddr proto= proto tag1 tag2 tag3 Interface= ifcaddr (dir) secclass= secclass dest= dest len= len vpnaction= vpnaction tunnelID= tunID ifcname= ifcname fragment= frag

Explanation

An IP packet matched the indicated filter rule but the tunnel is not active. For this record to be written, the matched filter rule must have IpFilterLogging set to yes.

timestamp is the stack timestamp that indicates the time at which the IP packet was processed by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

rulename is the filter rule name. If the IP packet matched a dynamic filter rule, the rule name of the corresponding anchor filter rule will be displayed; otherwise, the rule name of the matching filter rule will be displayed.

instance is the rule name extension that indicates which instance of the rule name was matched.

sipaddr is the source IP address.

dipaddr is the destination IP address.

proto is the protocol from the packet. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
- AH(51)
- ICMPv6(58)
- OSPF(89)
- IPIP(94)
- MIPv6(135)
- Unknown
- The protocol number

The tag1 value varies depending on the proto value.

- If the proto value is ICMP or ICMPv6, the tag1 value is type= followed by the ICMP or ICMPv6 type, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the proto value is TCP or UDP, the tag1 value is sport= followed by the source port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
- If the proto value is OSPF, the tag1 value is type= followed by the type, or followed by the value Unknown if the OSPF header is not present in the packet as the result of fragmentation.
- If the proto value is MIPv6, the tag1 value is type= followed by the type, or followed by the value Unknown if the MIPv6 header is not present in the packet as the result of fragmentation.
• If the proto value is any value not previously mentioned, the tag1 value is - which indicates that the data is not applicable.

**tag2 value varies depending on the proto value.**

• If the proto value is ICMP or ICMPv6, the tag2 value is code= followed by the ICMP or ICMPv6 code, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
• If the proto value is TCP or UDP, the tag2 value is dport= followed by the destination port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
• If the proto value is any value not previously mentioned, the tag2 value is - which indicates that the data is not applicable.

**tag3 value varies depending on the proto value and direction.**

• If the proto value is TCP or UDP, the direction is inbound, and the port has been translated by the CommServer NAT Traversal function, the tag3 value is origport= followed by the original source port.
• If the proto value is TCP or UDP, the direction is outbound, and the port has been translated by the CommServer NAT Traversal function, the tag3 value is origport= followed by the original destination port.
• If the proto value is any value not previously mentioned, the tag3 value is - which indicates that the data is not applicable.

ifcaddr is the interface address over which the packet was received or sent.

dir is I if packet is inbound, O if packet is outbound.

secclass is the security class assigned to the interface. Security class is a numeric value in the range of 0–255.

dest is local if a local destination or routed if being routed.

len is the packet length.

vpnaction is applicable if a VpnAction name is associated with the matched filter. Otherwise, N/A will be shown. If the tunnel is a manual tunnel, this is the name specified on the IpManVpnAction statement. If the tunnel is a dynamic tunnel, this is the name specified on the IpDynVpnAction statement.

tunID is the tunnel ID.

ifcname is the interface name

frag specifies whether the packet is a fragment. The value is Y if the packet is a fragment, or N if the packet is not a fragment.

**System action**

TCP/IP processing continues.

**Operator response**

Contact the system programmer.

**System programmer response**

If the indicated tunnel should be active, use the ipsec command to activate the tunnel. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the man ipsec command in a z/OS UNIX shell to obtain information about the ipsec command syntax and options.

**User response**

Not applicable.

**Problem determination**

Not applicable.
Source

z/OS Communications Server TCP/IP: TRMD

Module

EZATRZOS

Routing code

Descriptor code

Automation

Not applicable.

Example

| EZD0822I | Packet denied, tunnel inactive: 07/05/2007 16:19:44.39 filter rule= ipsec-2 ext= 1 sipaddr= 9.42.130.185 dipaddr= 10.1.1.1 proto= tcp(6) sport= 80 dport= 1026 -= Interface= 9.1.1.1 (O) secclass= 255 dest= local len= 284 vpnaction= ManualAction tunnelID= M4 ifcname= TRLE1AL fragment= N |

| EZD0823I | UDP Encapsulated ESP packet can not be routed: timestamp sipaddr= sipaddr dipaddr= dipaddr proto= proto vpnaction= vpnaction tunnelID= tunID ESPSPI= ESPIndex |

Explanation

The final destination address of a UDP-encapsulated ESP packet is not local. Routing beyond the tunnel endpoint is not supported for this type of encapsulation.

timestamp is the stack timestamp that indicates the time at which the failure was detected by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time. This timestamp might be different than the syslogd message timestamp.

sipaddr is the public source IP address.

dipaddr is the destination IP address.

proto is the protocol from the decapsulated packet. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- OSPF(89)
- IPIP(94)

- The protocol number

vpnaction is the name specified on the IpDynVpnAction statement.

tunID is the tunnel ID.

ESPIndex is the ESP security parameter index.

System action

The packet is dropped and TCP/IP processing continues.
Operator response

Contact the system programmer.

System programmer response

Ensure that the tunnel is defined correctly on the sending and receiving systems. See the information about IP Security in z/OS Communications Server: IP Configuration Guide for information about defining IPSec tunnels.
Use the `ipsec` command to display filter and tunnel information. See the information about managing network security in z/OS Communications Server: IP System Administrator’s Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

Module

EZATRZOS

Procedure name

trmd_ipsec_log

EZD0824I Possible NAT traversal remapping detected: timestamp original sipaddr=origsipaddr new sipaddr= newsipaddr original sport=origport new sport=newport dipaddr=dipaddr proto=proto vpnaction= vpnaction tunnelID= tunID ESPSPI= tunID

Explanation

A packet received over the specified tunnel contained a source IP address or source port that was different than the value at the time the tunnel was negotiated. If `origsipaddr` does not match `newsipaddr`, an address remapping might have occurred at the remote network address translation (NAT) device. If `origport` does not match `newport`, a port remapping might have occurred at the remote network address port translation (NAPT) device.

`timestamp` is the stack timestamp that indicates the time at which the failure was detected by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time. This timestamp might be different than the syslogd message timestamp.

`origsipaddr` is the IP address of the tunnel current remote endpoint.

`newsipaddr` is the source IP address from the inbound packet.

`origport` is the remote IKE peer port at the time the tunnel was negotiated.

`newport` is the remote IKE peer port from the UDP encapsulation header of the inbound packet.

`dipaddr` is the destination IP address from the inbound packet.

`proto` is the protocol from the decapsulated packet. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- OSPF(89)
- IPIP(94)
- The protocol number

`vpnaction` is the name specified on the IpDynVpnAction statement.

`tunID` is the tunnel ID.

`tunID` is the ESP security parameter index.
System action
The current inbound packet is dropped and processing is initiated to verify whether a NAT remapping actually occurred. Subsequent packets that do not match the tunnel current remote endpoint of the IKE peer port are also dropped. TCP/IP processing continues.

Operator response
None.

System programmer response
None.

Module
EZATRZOS

Procedure name
trmd_ipsec_log

EZD0825I  UDP encapsulated ESP unsupported protocol: timestamp sipaddr= sipaddr dipaddr= dipaddr proto= proto vpnaction= vpnaction tunnelID= tunID ESPSPI= ESPIndex

Explanation
A UDP-encapsulated ESP packet was decapsulated, and the resulting packet contained a protocol other than TCP, UDP or ICMP. No other protocols are supported for a UDP-encapsulated ESP tunnel when the remote tunnel endpoint is a security gateway behind a NAT.

timestamp is the stack timestamp that indicates the time at which the IP packet was processed by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time. This timestamp might be different than the syslogd message timestamp.

sipaddr is the public source IP address.

dipaddr is the destination IP address.

proto is the protocol from the decapsulated packet. Possible values are:

- IGMP(2)
- IP(4)
- OSPF(89)
- IPIP(94)
- The protocol number

vpnaction is the name specified on the IpDynVpnAction statement.

tunID is the tunnel ID.

ESPIndex is the ESP security parameter index.

System action
The packet is discarded and TCP/IP processing continues.

Operator response
None.
System programmer response

None.

Module

EZATRZOS

Procedure name

trmd_ipsec_log

EZD0826I  Remote port translation failed: timestamp sipaddr= sipaddr dipaddr= dipaddr proto= proto srcport=srcport dstport=dstport ikeport=ikeport vpnaction= vpnaction tunnelID= tunID ESPSPI= ESPIndex rsn= rsn

Explanation

An unsuccessful attempt was made to perform remote port translation. The rsn value provides additional information about the failure.

timestamp is the stack timestamp that indicates the time at which the failure was detected by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time. This timestamp might be different than the syslogd message timestamp.

sipaddr is the public source IP address.

dipaddr is the destination IP address.

proto is the protocol from the decapsulated packet. Possible values are:

- TCP(6)
- UDP(17)

srcport is the connection source port.

dstport is the connection destination port.

ikeport is the source port from the UDP encapsulation header.

vpnaction is the name specified on the IpDynVpnAction statement.

tunID is the tunnel ID.

ESPIndex is the ESP security parameter index.

rsn is the reason code. Possible values are:

1  The connection source port (srcport) in the inbound packet was already in use by another client with the same public source IP address. No alternate port was available.

2  A storage shortage prevented an alternate port from being assigned.

3  An internal error occurred during table lookup.

4  An internal error prevented the port translation entry from being added.

6  An internal error occurred when port translation was requested for a protocol that was not valid.

7  An internal error prevented the port translation entry from being added.
System action
The packet is dropped and TCP/IP processing continues.

Operator response
Contact the system programmer.

System programmer response
The value of rsn determines the appropriate system programmer response.

1
The translated port selection is limited to the port range specified in the filter policy that the packet matches. If a translated port could not be assigned, then the maximum number of ports specified in the filter policy are in use. Use the `ipsec` command to display the filter and the port translation entries.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

See the information about remote port translation in z/OS Communications Server: IP Configuration Guide for more information about port translation.

2
Determine the cause of the storage shortage. See z/OS Communications Server: IP Diagnosis Guide for information about storage shortages.

3
If this message appears repeatedly, take a dump of TCP/IP and contact the IBM Software Support Center.

4
If this message appears repeatedly, take a dump of TCP/IP and contact the IBM Software Support Center.

5
If this message appears repeatedly, take a dump of TCP/IP and contact the IBM Software Support Center.

6
If this message appears repeatedly, take a dump of TCP/IP and contact the IBM Software Support Center.

7
If this message appears repeatedly, take a dump of TCP/IP and contact the IBM Software Support Center.

Module
EZATRZOS

Procedure name
trmd_ipsec_log

EZD0827I  Remote port translated: timestamp sipaddr= sipaddr dipaddr= dipaddr proto= proto srcport=srcport dstport=dstport ikeport=ikeport xlateport=xlateport vpnaction= vpnaction tunnelID= tunID ESPSPI= ESPIndex

Explanation
The connection source port in the inbound packet (srcport) was already in use by another client with the same public source IP address and a translated source port for this client was not already assigned. A translated source port (xlateport) is assigned and will be used to complete the connection. Subsequent packets from this client using the same source port will also use the translated port value specified by xlateport.

timestamp is the stack timestamp that indicates the time at which the failure was detected by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time. This timestamp might be different than the syslogd message timestamp.

sipaddr is the public source IP address.
dipaddr is the destination IP address at the time the translated source port was assigned. Subsequent packets using the translated port (xlateport) might have a different destination IP address.

proto is the protocol from the decapsulated packet. Possible values are:

- TCP(6)
- UDP(17)

srcport is the original connection source port.

dstport is the connection destination port at the time the translated source port was assigned. Subsequent packets using the translated port (xlateport) might have a different destination port.

ikeport is the source port from the UDP encapsulation header.

xlateport is the port assigned by IPSec processing that will be used on this stack in place of srcport.

vpnaction is the name specified on the IpDynVpnAction statement.

tunID is the tunnel ID.

ESPindex is the ESP security parameter index.

**System action**
TCP/IP processing continues.

**Operator response**
None.

**System programmer response**
None. This message is for informational purposes only. If it is necessary to obtain information about this connection from the client system, the original connection source port might be needed. The original source port can be found in this log message or by using the `ipsec` command to display the translated port information.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

See the information about remote port translation in z/OS Communications Server: IP Configuration Guide for additional information about port translation.

**Module**
EZATRZOS

**Procedure name**
trmd_ipsec_log

**EZD0828I**
TCP remote port translation mismatch:timestamp sipaddr= sipaddr dipaddr= dipaddr srcport=srcport dstport=dstport origport=origport originst =originst currinst=currinst

**Explanation**
The remote port translation information at the time the connection was established does not match the information assigned to the current inbound packet. This can occur if the connection was initiated in the clear but is currently using a UDP-encapsulated tunnel or vice versa. If this is the case, the value of `originst` or `currinst` will be 0.

timestamp is the stack timestamp that indicates the time at which the failure was detected by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time. This timestamp might be different than the syslogd message timestamp.
sipaddr is the public source IP address.

dipaddr is the destination IP address.

srcport is the translated connection source port.

dstport is the connection destination port.

origport is the original connection source port from the inbound packet.

originst is the translated port instance ID at the time the connection was established.

currinst is the translated port instance ID assigned to the current packet.

**System action**
The packet is dropped and TCP/IP processing continues.

**Operator response**
Restart the TCP connection.

**System programmer response**
None.

**Module**
EZATRZOS

**Procedure name**
trmd_ipsec_log

**EZD0829I** Connections for DVIPA ip_address can not be recovered: timestamp

**Explanation**
The dynamic virtual IP address (DVIPA), which is one of the endpoints of a sysplex-wide Security Association (SWSA), is being taken over by another TCPIP stack. Any connection using a UDP-encapsulated tunnel, in which the IKE daemon can act only as the responder, for either the phase 1 SA or the phase 2 SA, cannot be recovered. Because the IKE daemon can act only as the responder, the UDP encapsulated tunnel cannot be recovered by the backup stack.

ip_address is the dynamic virtual IP address (DVIPA).

timestamp is the stack timestamp that indicates the time at which the failure was detected by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

**System action**
The TCPIP stack taking over the DVIPA is not informed about the connections. TCP/IP processing continues.

**Operator response**
None.

**System programmer response**
None.
Module
EZATRZOS

Procedure name
trmd_ipsec_log

EZD0830I Tunnel distribution failed: timestamp vpnaction=vpnaction tunnel ID= tunID AHSPI=AHIndex ESPSPI=ESPIndex

Explanation
The specified tunnel cannot be distributed because an error occurred while initializing data in the coupling facility.

timestamp is the stack timestamp that indicates the time at which the tunnel was added by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time. This timestamp might be different than the syslogd message timestamp.

vpnaction is the vpnaction name specified on the IpDynVpnAction statement.

tunID is the tunnel ID.

AHIndex is the AH security parameter index.

ESPIndex is the ESP security parameter index.

System action
TCP processing continues; the tunnel is not distributed but might be used on this system.

Operator response
Contact the system programmer. When the issue is resolved, refresh the indicated SA to allow distribution.

System programmer response
Ensure that the sysplex-wide Security Association (SWSA) coupling facility structure is set up correctly.

See the information about diagnosing sysplex-wide Security Association (SWSA) problems in z/OS Communications Server: IP Diagnosis Guide for information about diagnosing SWSA problems.

Module
EZATRMD

Procedure name
trmd_ipsec_log

EZD0831I Tunnel takeover preparation failed: timestamp vpnaction=vpnaction tunnel ID= tunID AHSPI=AHIndex ESPSPI=ESPIndex

Explanation
The specified tunnel cannot be taken over because an error occurred while storing data into the coupling facility.

timestamp is the stack timestamp that indicates the time at which the tunnel was added by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time. This timestamp might be different than the syslogd message timestamp.

vpnaction is the vpnaction name specified on the IpDynVpnAction statement.

tunID is the tunnel ID.
AHindex is the AH security parameter index.

ESPindex is the ESP security parameter index.

**System action**
TCP processing continues; the tunnel cannot be taken over but might be used on this system.

**Operator response**
Contact the system programmer. When the issue is resolved, refresh the indicated SA to allow takeover.

**System programmer response**
Ensure that the sysplex-wide Security Association (SWSA) coupling facility structure is set up correctly.

See the information about diagnosing sysplex-wide Security Association (SWSA) problems in z/OS Communications Server: IP Diagnosis Guide for information about diagnosing SWSA problems.

**Module**
EZATRMD

**Procedure name**
trmd_ipsec_log

**EZD0832I** Packet denied by NAT Traversal Processing: timestamp filter rule= rulename ext= instance sipaddr= sipaddr dipaddr= dipaddr proto= proto tag1 tag2 tag3 Interface= ifcaddr (dir) dest= dest len= len vpnaction=vpnaction rsn=rsn ifcname= ifcname fragment= frag

**Explanation**
An IP packet matched the indicated filter rule but further processing for NAT Traversal caused the packet to be denied. The rsn field provides more detailed information. For this message to be written, the matched filter rule must have IpFilterLogging set to yes.

timestamp is the stack timestamp that indicates the time at which the IP packet was denied by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

rulename is the anchor filter rule name. The value of N/A is displayed when a target stack is processing an inbound packet that was received on the distributing stack as a UDP-encapsulated ESP packet. The packet was decapsulated by the distributor before the distributor forwarded it to the target stack.

instance is the rule name extension that indicates which instance of the rule name was matched. The value of N/A is displayed when a target stack is processing an inbound packet that was received on the distributing stack as a UDP-encapsulated ESP packet. The packet was decapsulated by the distributor before the distributor forwarded it to the target stack.

sipaddr is the source IP address.

dipaddr is the destination IP address.

proto is the protocol from the packet. Possible values are:
- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
The tag1 value varies depending on the proto value:

- If the proto value is ICMP, the tag1 value is type= followed by the ICMP type, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the proto value is TCP or UDP, the tag1 value is sport= followed by the source port, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the proto value is OSPF, the tag1 value is type= followed by the type, or followed by the value Unknown if the ICMP header is not present in the packet due to fragmentation.
- If the proto value is MIPv6, the tag1 value is type= followed by the type, or followed by the value Unknown if the MIPv6 header is not present in the packet as the result of fragmentation.
- If the proto value is any value not previously mentioned, the tag1 value is -= which indicates that the data is not applicable.

Tag2 is one of the following:

- If the proto value is ICMP, the tag2 value is code= followed by the ICMP code, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the proto value is TCP or UDP, the tag2 value is dport= followed by the destination port, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the proto value is any value not previously mentioned, the tag2 value is -= which indicates that the data is not applicable.

Tag3 value varies depending on the proto value and direction:

- If the proto value is TCP or UDP, the direction is inbound, and the port has been translated by the Communications Server NAT Traversal function, the tag3 value is origport= followed by the original source port.
- If the proto value is TCP or UDP, the direction is outbound, and the port has been translated by the Communications Server NAT Traversal function, the tag3 value is origport= followed by the original destination port.
- If the proto value is any value not previously mentioned, the tag3 value is -= which indicates that the data is not applicable.

Ipfaddr is the interface address over which the packet was received or sent.

dir is I if packet is inbound, O if packet is outbound.

dest is local if a local destination or routed if being routed.

Len is the packet length.

Vpnaction is the name specified on the IpDynVpnAction statement for the referenced filter rule.

Rsn is the reason code that indicates the specific NAT Traversal processing error. The rsn is one of the following:

<table>
<thead>
<tr>
<th>rsn value</th>
<th>Affected packet</th>
<th>Explanation</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Inbound TCP or UDP packet.</td>
<td>An internal error occurred when attempting to create a NAT Resolution Filter.</td>
<td></td>
</tr>
<tr>
<td>rsn value</td>
<td>Affected packet</td>
<td>Explanation</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>2</td>
<td>Inbound TCP or UDP packet.</td>
<td>No storage could be allocated for a NAT Resolution Filter.</td>
<td>Storage to complete the request is not currently available. Until the storage shortage is relieved, packets will continue to be discarded.</td>
</tr>
<tr>
<td>3</td>
<td>Inbound TCP or UDP packet.</td>
<td>Unable to allocate a NAT Resolution Filter. The tunnel over which the packet was received cannot be found for the filter rule that the packet matched.</td>
<td>This could be the result of a policy mismatch between the peers. For example, an inbound packet that is received in the clear (for example, not encapsulated) but matches on a filter rule that specifies encapsulation.</td>
</tr>
<tr>
<td>4</td>
<td>Inbound non-TCP/UDP/ICMP packet</td>
<td>An inbound packet with a protocol not equal to TCP(6), UDP(17), or ICMP(1) matched on a NAT Traversal Anchor Filter.</td>
<td>When the IKE peer is a security gateway or the IKE peer is behind an NAPT, only inbound packets with a protocol value of TCP, UDP, or ICMP are supported over the UDP-encapsulated ESP tunnel.</td>
</tr>
<tr>
<td>5</td>
<td>Outbound TCP or UDP packet.</td>
<td>Unable to locate a matching NAT Resolution Filter.</td>
<td>When the IKE peer is a security gateway or the IKE peer is behind an NAPT, the NAT Resolution Filter is needed to determine which tunnel should be used for outbound packets. Data must be initiated from the client behind the security gateway or the client behind the NAPT.</td>
</tr>
<tr>
<td>6</td>
<td>Outbound non-TCP/UDP/ICMP packet</td>
<td>An outbound packet with a protocol not equal to TCP(6), UDP(17), or ICMP(1) matched on a NAT Traversal Anchor Filter.</td>
<td>When the IKE peer is a security gateway or the IKE peer is behind an NAPT, only outbound packets with a protocol value of TCP, UDP, or ICMP are supported over the UDP-encapsulated ESP tunnel.</td>
</tr>
<tr>
<td>7</td>
<td>Inbound ICMP packet</td>
<td>The tunnel over which the packet was received cannot be found for the filter rule that the ICMP packet matched.</td>
<td>This could be the result of a policy mismatch between the peers.</td>
</tr>
<tr>
<td>8</td>
<td>Outbound ICMP packet</td>
<td>Unable to locate the tunnel to use for the outbound packet. The outbound ICMP packet is not in response to an inbound packet.</td>
<td>When the IKE peer is a security gateway or the IKE peer is behind an NAPT, an outbound ICMP packet can be sent only over a UDP-encapsulated ESP tunnel in response to an inbound packet. For example, an Echo response can be sent in response to an Echo Request. Or an ICMP Port Unreachable message can be sent in response to an inbound UDP packet.</td>
</tr>
<tr>
<td>rsn value</td>
<td>Affected packet</td>
<td>Explanation</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| 9         | Outbound ICMP packet   | Unable to locate the tunnel to use for the outbound packet. The outbound ICMP packet cannot use the same tunnel as the inbound request. | When the IKE peer is a security gateway or the IKE peer is behind an NAPT, an outbound ICMP packet can be encapsulated and sent over a tunnel if the following are true:  
  • The outbound packet is in response to an inbound packet and,  
  • The tunnel used for the inbound packet can be used for the outbound packet.  
  If, for example, separate tunnels are negotiated for UDP and ICMP traffic, an outbound ICMP port unreachable packet cannot be sent over the same tunnel as the inbound UDP packet that triggered the ICMP outbound packet. When the IKE peer is a security gateway or the IKE peer is behind an NAPT and UDP-encapsulated ESP tunnels are being used, consideration should be given to using tunnels that encompass all protocols. |
| 10        | Inbound or outbound TCP packet | Unable to accept the TCP packet because the IPSec policy for the TCP connection has changed. The connection was initiated as clear text traffic but is now using a UDP-encapsulated tunnel or vice versa. | When a TCP connection traverses a NAT, the connection must be restarted after a filter policy change that causes the connection's traffic to change from IPSec-protected traffic to clear text, or from clear text to IPSec-protected traffic. |
| 11        | Outbound packet        | Unable to determine the local host public address for use in the IP header of the inner packet. | When the IKE peer is a security gateway and the NAT is in front of the local host, an outbound packet can be encapsulated and sent over a tunnel only if a packet has first been received inbound over the tunnel. Data must be initiated from the client behind the security gateway. |
| 12        | Inbound TCP or UDP packet | An internal error occurred when attempting to create a NAT Resolution Filter. |                                                                                                     |

*ifcname* is the interface name

*frag* specifies whether the packet is a fragment. The value is Y if the packet is a fragment, or N if the packet is not a fragment.

**System action**

The packet is dropped and TCP/IP processing continues.

**Operator response**

If the rsn value is 10, restart the TCP connection. Otherwise, contact the system programmer.
System programmer response

Unless a specific response is based on the rsn value shown in the following table, ensure that the filters and tunnel are defined correctly on the sending and receiving systems. Use the `ipsec` command to display filter and tunnel information.

<table>
<thead>
<tr>
<th>rsn value</th>
<th>System programmer response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Contact the IBM Software Support Center.</td>
</tr>
<tr>
<td>2</td>
<td>Determine the cause of the storage shortage. See z/OS Communications Server: IP Diagnosis Guide information about storage shortages.</td>
</tr>
</tbody>
</table>

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the ipsec command syntax and options.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP: TRMD

Module

EZATRZOS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

Example

EZD0832I Packet denied by NAT Traversal Processing: 07/05/2007 16:19:44.39 filter rule= ipsec-2 ext= 1 sipaddr= 9.42.130.185 dipaddr= 10.1.1.1 proto= tcp(6) sport= 1026 dport= 80 -= Interface= 9.1.1.1 (I) secclass= 255 dest= local len= 284 vpnaction= DynAction rsn= 4 ifcname= TRLE1AL fragment= N

Procedure name

trmd_ipsec_log

EZD0833I Packet denied, tunnel mismatch: timestamp filter rule= rulename ext= instance sipaddr= sipaddr dipaddr= dipaddr proto= proto tag1 tag2 tag3 Interface= ifcaddr (dir) dest= dest len= len tunnelID= tunID decap_tunnelID= decap_tunnelID ifcname= ifcname fragment= frag
Explanation
An inbound IP packet matched the indicated filter rule but was denied because the packet was not encapsulated as specified in the filter rule. For this message to be written, the matched filter rule must have IpFilterLogging set to yes or logdeny.

timestamp is the stack timestamp that indicates the time at which the IP packet was denied by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

rulename is the filter rule name. If the IP packet matched a dynamic filter rule, the rule name of the corresponding anchor filter rule will be displayed; otherwise, the rule name of the matching filter rule will be displayed.

• In the policy agent configuration file, rulename is the name specified on an IpFilterRule statement.
• When configured with the IBM Configuration Assistant for z/OS Communications Server rulename corresponds to the name of a Connectivity Rule in the GUI. rulename also contains a numeric suffix appended to the Connectivity Rule name to guarantee uniqueness.

instance is the rule name extension that indicates which instance of the rule name was matched.

sipaddr is the source IP address.

dipaddr is the destination IP address.

proto is the protocol from the packet. Possible values are:

• ICMP(1)
• IGMP(2)
• IP(4)
• TCP(6)
• UDP(17)
• ESP(50)
• AH(51)
• ICMPv6(58)
• OSPF(89)
• IPIP(94)
• MIPv6(135)
• Unknown
• The protocol number

The tag1 value varies depending on the proto value.

• If the proto value is ICMP or ICMPv6, the tag1 value is type= followed by the ICMP or ICMPv6 type, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
• If the proto value is TCP or UDP, the tag1 value is sport= followed by the source port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
• If the proto value is OSPF, the tag1 value is type= followed by the type, or followed by the value Unknown if the OSPF header is not present in the packet as the result of fragmentation.
• If the proto value is MIPv6, the tag1 value is type= followed by the type, or followed by the value Unknown if the MIPv6 header is not present in the packet as the result of fragmentation.
• If the proto value is any value not previously mentioned, the tag1 value is -= which indicates that the data is not applicable.

The tag2 value varies depending on the proto value.

• If the proto value is ICMP or ICMPv6, the tag2 value is code= followed by the ICMP or ICMPv6 code, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
• If the \texttt{proto} value is TCP or UDP, the \texttt{tag2} value is \texttt{dport=} followed by the destination port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.

• If the \texttt{proto} value is any value not previously mentioned, the \texttt{tag2} value is \texttt{=} which indicates that the data is not applicable.

\texttt{tag3} value varies depending on the \texttt{proto} value and direction.

• If the \texttt{proto} value is TCP or UDP, the direction is inbound, and the port has been translated by the CommServer NAT Traversal function, the \texttt{tag3} value is \texttt{origport=} followed by the original source port.

• If the \texttt{proto} value is TCP or UDP, the direction is outbound, and the port has been translated by the CommServer NAT Traversal function, the \texttt{tag3} value is \texttt{origport=} followed by the original destination port.

• If the \texttt{proto} value is any value not previously mentioned, the \texttt{tag3} value is \texttt{=} which indicates that the data is not applicable.

\texttt{ifcaddr} is the interface address over which the packet was received or sent.

\texttt{dir} is \texttt{I} if packet is inbound, \texttt{O} if packet is outbound.

\texttt{dest} is \texttt{local} if a local destination or \texttt{routed} if being routed.

\texttt{len} is the packet length.

\texttt{tunID} is the tunnel ID for the tunnel specified by the filter rule. A value of \texttt{N/A} indicates that the filter rule permits the IP packet without IPSec protection.

\texttt{decap\_tunID} is the tunnel ID for the tunnel used to decapsulate the IP packet. A value of \texttt{N/A} indicates that the IP packet was not IPSec encapsulated.

\texttt{ifcname} is the interface name

\texttt{frag} specifies whether the packet is a fragment. The value is \texttt{Y} if the packet is a fragment, or \texttt{N} if the packet is not a fragment.

\textbf{System action}

TCP/IP processing continues.

\textbf{Operator response}

Contact the system programmer.

\textbf{System programmer response}

Ensure that the filters and tunnel are defined correctly on the sending and receiving systems. Use the \texttt{ipsec} command to display filter and tunnel information. See the information about managing network security in \texttt{z/OS Communications Server: IP System Administrator's Commands} or issue the \texttt{man ipsec} command in a \texttt{z/OS UNIX} shell to obtain information about the \texttt{ipsec} command syntax.

\textbf{Module}

\texttt{EZATRZOS}

\textbf{Example}

\begin{verbatim}
EZD0021I Packet denied, no tunnel: 07/05/2007 16:19:44.39 filter rule= ipsec-2 ext= 1 sipaddr= 9.42.130.185 dipaddr= 10.1.1.1 proto= tcp(6) sport= 80 dport= 1026 -= Interface= 9.1.1.1 (O) secclass= 255 dest= local len= 284 vpnaction= DynAction ifcname= TRLE1AL fragment= N
\end{verbatim}

\textbf{Procedure name}

\texttt{trmd\_ipsec\_log}
Encapsulation failed: timestamp sipaddr= sipaddr dipaddr= dipaddr proto= proto vpnaction= vpnaction tunnelID= tunID rsn=rsn ICSF Return Code= return_code ICSF Reason Code = reason_code AHSPI= AHIndex ESPSPI= ESPIndex

Explanation
The IPSec packet cannot be encapsulated and is discarded.

In the message text:

*timestamp*
Indicates when the encapsulation failure occurred. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

*sipaddr*
The source IP address.

*dipaddr*
The destination IP address.

*proto*
The protocol. Possible values are:
- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ICMPv6(58)
- OSPF(89)
- IPIP(94)
- The protocol number

*vpnaction*
The vpnaction name.
- If configured with the IBM Configuration Assistant for z/OS Communications Server, the vpnaction name corresponds to the name of the security level in the GUI. The vpnaction name also contains a suffix that is appended to the security level name to guarantee uniqueness.
- If configured in the Policy Agent configuration file, the vpnaction value is one of the following:
  - If the tunnel is a manual tunnel, the vpnaction value is the name specified on the IpManVpnAction statement.
  - If the tunnel is a dynamic tunnel, the vpnaction value is the name specified on the IpDynVpnAction statement. If a tunnel is not found, the vpnaction value is N/A.

*tunID*
The tunnel ID. If the vpnaction value is N/A, a tunnel with matching end points and security parameter indices (spi) could not be found.

*rsn*
The specific reason encapsulation failed. The rsn value is one of the following:

<table>
<thead>
<tr>
<th>rsn value</th>
<th>Explanation</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Encryption error</td>
<td>An error occurred while trying to encrypt an outbound packet.</td>
</tr>
<tr>
<td>rsn value</td>
<td>Explanation</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>2</td>
<td>AH authentication error</td>
<td>An error was encountered when trying to authenticate an outbound packet. This problem might also be caused by a failure in an ICSF service. If so, the specific failure will be reported on the ICSF Return Code and ICSF Reason Code fields.</td>
</tr>
<tr>
<td>3</td>
<td>ESP authentication error</td>
<td>An error was encountered when trying to authenticate an outbound packet. This problem might also be caused by a failure in an ICSF service. If so, the specific failure will be reported on the ICSF Return Code and ICSF Reason Code fields.</td>
</tr>
<tr>
<td>4</td>
<td>Maximum packet exceeded</td>
<td>The addition of ESP headers will cause the maximum packet size to be exceeded.</td>
</tr>
<tr>
<td>5</td>
<td>Lifesize exceeded</td>
<td>The number of bytes in the outbound packet will cause the lifesize specification to be exceeded for the tunnel.</td>
</tr>
<tr>
<td>6</td>
<td>Unknown authentication algorithm</td>
<td>An internal error occurred while authenticating the packet.</td>
</tr>
<tr>
<td>7</td>
<td>Unknown encryption algorithm</td>
<td>An internal error occurred while encrypting the packet.</td>
</tr>
<tr>
<td>8</td>
<td>No tunnel found</td>
<td>A tunnel was not found for the specified tunnel ID.</td>
</tr>
<tr>
<td>9</td>
<td>Sequence numbers were not obtained</td>
<td>Sequence numbers could not be obtained from the coupling facility for a distributed tunnel.</td>
</tr>
<tr>
<td>10</td>
<td>IP header not valid</td>
<td>The IP header of the packet being encapsulated does not contain the same source and destination IP address as specified in the tunnel; transport mode is in effect.</td>
</tr>
<tr>
<td>13</td>
<td>Storage shortage</td>
<td>Storage to complete the request is not currently available. Until the storage shortage is relieved, encapsulation will fail.</td>
</tr>
<tr>
<td>24</td>
<td>Encryption failure in ICSF Service</td>
<td>The ICSF Return Code and the ICSF Reason Code fields contain the return and reason codes that were returned from the ICSF service.</td>
</tr>
<tr>
<td>25</td>
<td>ICSF is not available</td>
<td>Either ICSF is not active or it has not completed initialization.</td>
</tr>
<tr>
<td>26</td>
<td>Version mismatch</td>
<td>The IP version of the packet did not match the IP version of the tunnel.</td>
</tr>
<tr>
<td>27</td>
<td>Encapsulation using transport mode</td>
<td>Encapsulation using transport mode was requested but the packet that is being processed is a routed packet. This is most likely the result of a policy definition error.</td>
</tr>
<tr>
<td></td>
<td>not valid for routed traffic</td>
<td>For manual tunnels, this might occur if routed traffic matches a filter rule referencing an IpManVpnAction statement that specified the transport method HowToEncap. For IPv6, this might occur if a routing header contains an intermediate hop that routed the packet back through the packet’s originating system. The tunnel endpoints matched the packet; however, the packet has been routed to this system.</td>
</tr>
<tr>
<td>rsn value</td>
<td>Explanation</td>
<td>Comments</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>28</td>
<td>Sequence number wrapped</td>
<td>The sequence number has wrapped, which indicates that the tunnel has expired. The tunnel will be deleted.</td>
</tr>
<tr>
<td>29</td>
<td>Sequence numbers were not obtained</td>
<td>Sequence numbers could not be obtained from the coupling facility for a distributed tunnel because a list entry is not allocated for the tunnel.</td>
</tr>
<tr>
<td>30</td>
<td>Sequence numbers were not obtained</td>
<td>Sequence numbers could not be obtained from the coupling facility for a distributed tunnel because VTAM is not active.</td>
</tr>
</tbody>
</table>

**return_code**
The return code value, in hexadecimal format, returned from the ICSF service.

**reason_code**
The reason code value, in hexadecimal format, returned from the ICSF service.

**AHIndex**
The AH security parameter index. If the failure occurred before the AH SPI was known, then n/a is displayed.

**ESPIndex**
The ESP security parameter index. If the failure occurred before the ESP SPI was known, then n/a is displayed.

**System action**
The packet is discarded and TCP/IP processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
The response is based on the rsn value, as shown in the following table.

<table>
<thead>
<tr>
<th>rsn value</th>
<th>System programmer response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 2, 3</td>
<td>If the problem is transient, no action is required. For manual tunnels, verify that the security parameters and encryption keys on the IpManVpnAction statement are correctly defined. When configured with the IBM Configuration Assistant for z/OS Communications Server, the IpManVpnAction name corresponds to the name of the security level in the GUI. The IpManVpnAction name also contains a suffix that is appended to the security level name to guarantee uniqueness. For more details about the IpManVpnAction statement, see the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Otherwise, ensure that the tunnel is defined correctly on the sending and receiving systems. Use the <strong>ipsec</strong> command to display filter and tunnel information. See the information about managing network security in z/OS Communications Server: IP System Administrator’s Commands or issue the <strong>man ipsec</strong> command in a z/OS UNIX shell to obtain information about the <strong>ipsec</strong> command syntax and options. If the problem was due to an ICSF failure, then see the information about the ICSF and TSS Return and Reason Codes in z/OS Cryptographic Services ICSF Application Programmer’s Guide for the specific actions to be taken.</td>
</tr>
<tr>
<td>4</td>
<td>Contact the IBM Software Support Center.</td>
</tr>
<tr>
<td>rsn value</td>
<td>System programmer response</td>
</tr>
<tr>
<td>-----------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>5, 28</td>
<td>If the problem is transient, no action is required. Otherwise, ensure that the tunnel is defined and activated correctly on the sending and receiving systems. Use the <code>ipsec</code> command to display filter and tunnel information. The <code>ipsec</code> command can also be used to refresh or activate a tunnel. See the information about managing network security in z/OS Communications Server: IP System Administrator’s Commands or issue the <code>man ipsec</code> command in a z/OS UNIX shell to obtain information about the <code>ipsec</code> command syntax and options.</td>
</tr>
<tr>
<td>6, 7, 8</td>
<td>Contact the IBM Software Support Center.</td>
</tr>
<tr>
<td>9, 29, 30</td>
<td>If the problem is transient, no action is required. Verify that z/OS VTAM is active and that the Coupling Facility is available and connected.</td>
</tr>
<tr>
<td>10</td>
<td>Contact the IBM Software Support Center.</td>
</tr>
<tr>
<td>13</td>
<td>Determine the cause of the storage shortage.</td>
</tr>
<tr>
<td>24</td>
<td>See the information about the ICSF and TSS Return and Reason Codes in z/OS Cryptographic Services ICSF Application Programmer’s Guide for the specific actions to be taken.</td>
</tr>
<tr>
<td>25</td>
<td>Start ICSF if it is not active.</td>
</tr>
<tr>
<td>26</td>
<td>Contact the IBM Software Support Center.</td>
</tr>
<tr>
<td>27</td>
<td>Ensure that the IPSec policy is defined correctly. Ensure that the filter rules for routed traffic do not reference VPN actions that request transport mode. Use the <code>ipsec</code> command to display filter and tunnel information. See the information about managing network security in z/OS Communications Server: IP System Administrator’s Commands or issue the <code>man ipsec</code> command in a z/OS UNIX shell to obtain information about the <code>ipsec</code> command syntax.</td>
</tr>
</tbody>
</table>

User response
Not applicable.

Problem determination
Not applicable.

Module
EZATRMD

EZD0835I IPv6 filters defined but IPv6 IPSECURITY support not enabled

Explanation
IPv6 IP filters were defined in the Policy Agent IP filter policy, but IPv6 IPSECURITY is not enabled on the IPCONFIG6 statement in the TCP/IP profile.

System action
IPv6 IP filters are discarded and IPv6 traffic is not subject to IP filtering. TCP/IP processing continues.

Operator response
Contact the system programmer.

System programmer response
Determine whether IPv6 IP filtering is necessary. If not, no action is necessary. If IPv6 IP filtering is necessary, enable it by coding the IPSECURITY option on the IPCONFIG6 statement in the TCP/IP profile. Restart the
TCP/IP stack for this change to take effect. See the information about the IPCONFIG6 statement in the z/OS Communications Server: IP Configuration Reference for more information about specifying IP security for a stack.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IPSec

**Module**
ezatrzos.c

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Example**
Not applicable.

**EZD0836I**
Packet permitted: timestamp sipaddr= sipaddr dipaddr= dipaddr proto= proto type= type code= code Interface= ifcaddr (dir) secclass= secclass dest= dest len= len tunnelID= tunID ifcname= ifcname embsipaddr= embsipaddr embdipaddr= embdipaddr embproto= embproto type= type tag1 tag2 tag3

**Explanation**
An ICMP error packet did not match a filter rule but was permitted based on information in the embedded packet that caused the error. For an inbound packet, the packet was received over a tunnel and the selectors in the embedded headers matched the selectors for the tunnel. For an outbound packet, the original packet for which the ICMP message was generated was received over a tunnel and the outbound ICMP packet will be encapsulated using this tunnel. For this message to be issued, the IPFilterPolicy policy must have filter logging enabled.

In the message text:

- **timestamp**
  The stack timestamp that indicates the time at which the IP packet was handled by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

- **sipaddr**
  The source IP address.

- **dipaddr**
  The destination IP address.

- **proto**
  The protocol from the packet. Possible values are:
  - ICMP(1)
- ICMPv6(58)

**type**
The ICMP or ICMPv6 type.

**code**
The ICMP or ICMPv6 code.

**ifcaddr**
The interface address over which the packet was received or sent.

**dir**
The possible values are **I** if the packet is inbound or **O** if the packet is outbound.

**secclass**
The security class assigned to the interface. Security class is a numeric value in the range 0 - 255.

**dest**
The possible values are **local** if the destination is local or **routed** if the destination is being routed.

**len**
The packet length.

**tunID**
The tunnel ID.

**ifcname**
The interface name.

**embsipaddr**
The source IP address from the embedded IP header.

**embdipaddr**
The destination IP address from the embedded IP header.

**embproto**
The protocol from the embedded IP header. Possible values are:
- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
- AH(51)
- ICMPv6(58)
- OSPF(89)
- IPIP(94)
- MIPV6(135)
- Unknown
- The protocol number

**tag1**
The **tag1** value varies depending on the **proto** value.

- If the **proto** value is ICMP or ICMPv6, the **tag1** value is **type=** followed by the ICMP or ICMPv6 type.
- If the **proto** value is TCP or UDP, the **tag1** value is **sport=** followed by the source port.
- If the **proto** value is OSPF, the **tag1** value is **type=** followed by the type.
- If the **proto** value is MIPv6, the **tag1** value is **type=** followed by the type.
- If the **proto** value is any value not previously mentioned, the **tag1** value is **-** which indicates that the data is not applicable.
**tag2**

tag2 value varies depending on the *proto* value.

- If the *proto* value is ICMP or ICMPv6, the *tag2* value is **code**= followed by the ICMP or ICMPv6 code.
- If the *proto* value is TCP or UDP, the *tag2* value is **dport**= followed by the destination port.
- If the *proto* value is any value not previously mentioned, *tag2* is -= which indicates that the data is not applicable.

**tag3**

tag3 value varies depending on the *proto* value and direction.

- If the *proto* value is TCP or UDP, the direction is inbound, and the port has been translated by the CommServer NAT Traversal function, the *tag3* value is **origport**= followed by the original source port.
- If the *proto* value is TCP or UDP, the direction is outbound, and the port has been translated by the CommServer NAT Traversal function, the *tag3* value is **origport**= followed by the original destination port.
- If the *proto* value is any value not previously mentioned, the *tag3* value is -= which indicates that the data is not applicable.

**System action**

TCP/IP processing continues.

**Operator response**

None.

**System programmer response**

None

**User response**

Not applicable.

**Problem determination**

None

**Source**

z/OS Communications Server TCP/IP: TRMD

**Module**

EZATRZOS

**Routing code**

Not applicable for syslog message.

**Descriptor code**

Not applicable for syslog message.

**Automation**

Not applicable
Example

EZD0836I Packet permitted: 09/11/2007 15:23:06.95 sipaddr= 10.11.2.4 dipaddr= 10.81.2.2 proto= icmp(1)
   type= 3 code= 1 Interface= 10.11.2.4 (O) secclass= 255 dest= local len= 56 tunnelID= Y4
   ifcname= MPC4124L embsipaddr= 10.81.2.2 embdipaddr= 10.81.8.8 embproto= udp(17) sport= 1050
dport= 10173 -=

EZD0837I Defensive filter packet denied messages limited: date time filter_rule= rulename filter_ext=
   instance filter_sipaddr= sipaddr / sip_prefix_length filter_dipaddr= dipaddr / dip_prefix_length
   filter_proto= proto tag1 tag2 filter_fragmentsonly= fragments_only filter_dir= dir
   filter_routing= routing suppressed_count= count

Explanation

This message is issued when limiting of filter match messages was requested for a defensive filter and at least
one "packet denied" message (EZD1721I) for the defensive filter was suppressed during the preceding five
minutes.

In the message text:

date
The date on which this message was issued. This date is retrieved from the system time-of-day clock, which
usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd
message timestamp.

time
The time at which this message was issued. This time is retrieved from the system time-of-day clock, which
usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd
message timestamp.

rulename
The defensive filter rule name as specified on the -N option when the defensive filter was added with the
z/OS UNIX ipsec command.

instance
The rule name extension.

sipaddr / sip_prefix_length
The source IP address specification for the defensive filter rule. The value 0.0.0.0/0 indicates that the
defensive filter rule applies to all source IPv4 addresses. The value ::/0 indicates that the defensive filter rule
applies to all source IPv6 addresses.

dipaddr / dip_prefix_length
The destination IP address specification for the defensive filter rule. The value 0.0.0.0/0 indicates that the
defensive filter rule applies to all destination IPv4 addresses. The value ::/0 indicates that the defensive filter
rule applies to all destination IPv6 addresses.

proto
The protocol specification for the defensive filter rule. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
- AH(51)
- ICMPv6(58)
- OSPF(89)
- IPIP(94)
- MIPv6(135)
• The protocol number
• ALL

**tag1**
The *tag1* value varies depending on the *proto* value.

If the *proto* value is ICMP or ICMPv6, the *tag1* value is `type=` followed by the ICMP or ICMPv6 type, or followed by the value all.

If the *proto* value is TCP or UDP, the *tag1* value is `sport=` followed by the source port range. For example, `sport= 1024 - 65535`. For a defensive filter that applies to all source ports the *tag1* value is `sport= 1 - 65535`.

If the *proto* value is any value not previously mentioned, the *tag1* value is `-=` which indicates that the data is not applicable.

**tag2**
The *tag2* value varies depending on the protocol.

If the *proto* value is ICMP or ICMPv6, the *tag2* value is `code=` followed by the ICMP or ICMPv6 code, or followed by the value all.

If the *proto* value is TCP or UDP, the *tag2* value is `dport=` followed by the destination port range. For example, `dport= 21 - 21`. For a defensive filter that applies to all destination ports, the *tag2* value is `dport= 1 - 65535`.

If the *proto* value is any value not previously mentioned, the *tag2* value is `-=` which indicates that the data is not applicable.

**fragments_only**
The fragment specification for the defensive filter rule. Possible values are:

• **yes** - The defensive filter rule applies only to fragments.
• **no** - The defensive filter rule does not apply only to fragments.

**dir**
The direction specified for the defensive filter rule. Possible values are inbound and outbound.

**routing**
The routing specified for the defensive filter rule. Possible values are local, routed, and either.

**count**
The number of "packet denied" messages (EZD1721I) for the defensive filter that were suppressed during the preceding five minutes.

**System action**
TCP/IP processing continues.

**Operator response**
No action is needed.

**System programmer response**
No action is needed.

**User response**
Not applicable.

**Problem determination**
Not applicable.
Source
z/OS Communications Server TCP/IP: TRMD

Module
EZATRZOS

Routing code
*

Descriptor code
*

Automation
Not applicable for automation.

Example

EZD0837I Defensive filter packet denied messages limited: 11/28/2011 16:35:55.42 filter_rule=Block_10_UDP_301 filter_ext= 1 filter sipaddr= 10.8.8.0 / 24 filter dipaddr= 0.0.0.0 / 0 filter proto= udp(17) sport= 301 - 301 dport= 1 - 65535 filter_fragmentsonly= no filter dir= inbound filter_routing= local suppressed_count= 125

EZD0838I Defensive filter packet would have been denied messages limited: date time filter_rule= rulename filter_ext= instance filter sipaddr= sipaddr / sip_prefix_length filter dipaddr= dipaddr / dip_prefix_length filter proto= proto tag1 tag2 filter fragmentsonly= fragments_only filter dir= dir filter_routing= routing suppressed_count= count

Explanation
This message is issued when limiting of filter match messages was requested for a defensive filter and at least one "packet would have been denied" message (EZD1722I) for the defensive filter was suppressed during the preceding five minutes.

In the message text:

date
The date on which this message was issued. This date is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

time
The time at which this message was issued. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

rulename
The defensive filter rule name as specified on the -N option when the defensive filter was added with the z/OS UNIX ipsec command.

instance
The rule name extension.

sipaddr/sip_prefix_length
The source IP address specification for the defensive filter rule. The value 0.0.0.0/0 indicates that the defensive filter rule applies to all source IPv4 addresses. The value ::/0 indicates that the defensive filter rule applies to all source IPv6 addresses.
**dipaddr/dip_prefix_length**

The destination IP address specification for the defensive filter rule. The value 0.0.0.0/0 indicates that the defensive filter rule applies to all destination IPv4 addresses. The value ::/0 indicates that the defensive filter rule applies to all destination IPv6 addresses.

**proto**

The protocol specification for the defensive filter rule. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
- AH(51)
- ICMPv6(58)
- OSPF(89)
- IPIP(94)
- MIPv6(135)
- The protocol number
- ALL

**tag1**

The tag1 value varies depending on the proto value.

If the proto value is ICMP or ICMPv6, the tag1 value is type= followed by the ICMP or ICMPv6 type, or followed by the value all.

If the proto value is TCP or UDP, the tag1 value is sport= followed by the source port range. For example, sport= 1024 - 65535. For a defensive filter that applies to all source ports the tag1 value is sport= 1 - 65535.

If the proto value is any value not previously mentioned, the tag1 value is -= which indicates that the data is not applicable.

**tag2**

The tag2 value varies depending on the protocol.

If the proto value is ICMP or ICMPv6, the tag2 value is code= followed by the ICMP or ICMPv6 code, or followed by the value all.

If the proto value is TCP or UDP, the tag2 value is dport= followed by the destination port range. For example, dport= 21 - 21. For a defensive filter that applies to all destination ports, the tag2 value is dport= 1 - 65535.

If the proto value is any value not previously mentioned, the tag2 value is -= which indicates that the data is not applicable.

**fragments_only**

The fragment specification for the defensive filter rule. Possible values are:

- yes - The defensive filter rule applies only to fragments.
- no - The defensive filter rule does not apply only to fragments.

**dir**

The direction specified for the defensive filter rule. Possible values are inbound and outbound.

**routing**

The routing specified for the defensive filter rule. Possible values are local, routed, and either.

**count**

The number of "packet would have been denied" messages (EZD1722I) for the defensive filter that were suppressed during the preceding five minutes.
**System action**
TCP/IP processing continues.

**Operator response**
No action is needed.

**System programmer response**
No action is needed.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: TRMD

**Module**
EZATRZOS

**Routing code**
*

**Descriptor code**
*

**Automation**
Not applicable for automation.

**Example**

```plaintext
EZD0838I Defensive filter packet would have been denied messages limited: 11/28/2011 16:35:55.42
filter_rule= Block_10 UDP_301 filter_ext= 1 filter_sipaddr= 10.8.8.0 / 24
filter_dipaddr= 0.0.0.0 / 0 filter_proto= udp(17) sport= 301 - 301 dport= 1 - 65535
filter_fragmentsonly= no
filter_dir= inbound filter_routing= local suppressed_count= 125
```

**EZD0839I**

Connection reset: date time filter rule= rulename ext= instance sipaddr= sipaddr dipaddr= dipaddr sport= sport dport= dport Interface= ifcaddr secclass= secclass ifcname= ifcname

**Explanation**
A TCP connection traversing a Shared Memory Communications over Remote Direct Memory Access (SMC-R) link matched the indicated filter rule and was reset. TCP connections that traverse SMC-R links and match a deny filter rule or a filter rule that specifies the use of IP security are reset. For this message to be written, the matched filter rule must have IpFilterLogging set to **yes**.

In the message text:
date
The date on which this message was issued. This date is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

time
The time at which this message was issued. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

rulename
The filter rule name. If the TCP connection matched a dynamic filter rule, the rule name of the corresponding anchor filter rule is displayed; otherwise, the rule name of the matching filter rule is displayed.

- In the policy agent configuration file, rulename is the name specified on the IpFilterRule statement.
- When configured with the IBM Configuration Assistant for z/OS Communications Server, rulename corresponds to the name of a Connectivity Rule in the GUI. rulename also contains a suffix appended to the Connectivity Rule name to guarantee uniqueness.

instance
The rule name extension that indicates which instance of the rule name was matched.

sipaddr
The source IP address of the TCP connection.

dipaddr
The destination IP address of the TCP connection.

sport
The source port of the TCP connection.

dport
The destination port of the TCP connection.

ifcaddr
The interface address over which the TCP connection was established.

secclass
The security class assigned to the interface. Security class is a numeric value in the range of 0 - 255.

ifcname
The name of the interface.

System action
TCP/IP processing continues.

Operator response
No action is needed.

System programmer response
No action is needed.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: TRMD

604  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Module
EZATRZOS

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD0839I Connection reset: 10/23/2012 15:40:17.54 filter rule= OSX_Rule ext= 2 sipaddr= 172.18.0.1
dipaddr= 172.18.0.2 sport = 8080 dport = 1026 Interface= 172.18.0.1 secclass= 255 ifcname= QDIOX4101

EZD0840I Claim list failed: date time structure name= structure_name

Explanation
An attempt to claim a list in the identified EZBDVIPA coupling facility structure failed because a list was not available.

In a sysplex, an EZBDVIPA coupling facility structure is required to support Sysplex-wide Security Associations (SWSA). Tunnel sequence numbers are stored in the EZBDVIPA structure to enable sysplex distribution. Tunnel data is stored in the EZBDVIPA structure to enable DVIPA takeover.

Depending on the number of DVIPAs and IPSec tunnels in use, TCP/IP can exhaust the number of lists defined in the EZBDVIPA structure. After all lists are claimed, subsequent attempts to claim a list fail, which causes data traffic over the affected tunnel to fail if the traffic is distributed. Also, the affected tunnel cannot be recovered after a DVIPA takeover.

In the message text

**date**
The date on which this message was issued. This date is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This date might be different than the syslogd message date.

**time**
The time at which this message was issued. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This time might be different than the syslogd message time.

**structure_name**
The name of the EZBDVIPA coupling facility structure for which the claim list failed.

**Guideline:** If subplexing is not in use, the name of the structure is EZBDVIPA. If you use subplexing within your sysplex, the name is in the form EZBDVIPAvvtt, where vv is the VTAM group ID and tt is the TCP/IP group ID.

**System action**
TCP/IP continues. Data traffic over the affected tunnel fails if the traffic is distributed. The affected tunnel cannot be recovered after a DVIPA takeover.

**Operator response**
Contact the system programmer.
**System programmer response**

Issue D NET,STATS,TYPE=VTAM,STRNAME=structure_name on each VTAM in the sysplex to determine the number of lists in the coupling facility structure.

If message IST1189I appears in the DISPLAY STATS output, this indicates that VTAM might not have access to all the lists. For additional information on the DISPLAY STATS output, see the description of IST1189I under the first message in the display, IST1370I. See Modifying the number of lists in z/OS Communications Server: SNA Network Implementation Guide for instructions on how to adjust the number of lists that VTAM can access in the EZBDVIPA structure.

If your configuration has a large number of DVIPAs and you expect a large number of tunnels between endpoints, see Modifying the number of lists in z/OS Communications Server: SNA Network Implementation Guide for instructions on how to increase the number of lists for the EZBDVIPA structure.

Otherwise, evaluate your policy definitions to identify the reason for the unexpectedly large number of tunnels.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZATRZOS

**Routing code**

Not applicable.

**Descriptor code**

Not applicable.

**Automation**

Not applicable.

**Example**

```
EZD0840I Claim list failed: 11/09/2016 14:01:49.06 structure name= EZBDVIPA
EZD0840I Claim list failed: 11/09/2016 14:01:49.06 structure name= EZBDVIPA0122
```

**EZD0851I**  Unable to open message catalog cat:: errno errno (description) errnojr errnojr - Default messages will be used

**Explanation**

An attempt was made to open the command message catalog cat in the message catalog directory, but the catalog could not be opened for the specified reason. The default location for the message catalog is set by the NLSPATH environment variable to be NLSPATH=/usr/lib/nls/msg/%L/%N.

`cat` is the name of the catalog the `ipsec` command attempted to open.
errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (erros) information in z/OS UNIX System Services Messages and Codes.

description describes the meaning of errno.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The **ipsec** command processing continues. Default messages will be used.

Operator response
If you want to use the message catalog, correct the indicated error. If the default messages are acceptable, no action is necessary.

System programmer response
If you want to use the message catalog, correct the indicated error. If the default messages are acceptable, no action is necessary.

Module
ipsec.c

Procedure name
command_init

**EZD0852I** Unknown option -opt

Explanation
Command parsing detected an unrecognized option.

opt is the unknown option.

System action
The **ipsec** command processing ends.

Operator response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the man **ipsec** command in a z/OS UNIX shell to obtain information about the **ipsec** command syntax and options.

System programmer response
None.

Module
ipsec.c

Procedure name
None.

**EZD0853I** Option -opt is missing required data
**Explanation**
The option identified by `opt` required input data, but none was specified on the command.
`opt` is the specified option.

**System action**
The `ipsec` command processing ends.

**Operator response**
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

**System programmer response**
None.

**Module**
`ipsec.c`

**Procedure name**
None.

---

**EZD0854I Cannot parse option opt**

**Explanation**
The system cannot determine how to parse `opt`.
`opt` is the option that could not be parsed.

**System action**
The `ipsec` command processing ends.

**Operator response**
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

**System programmer response**
None.

**Module**
`ipsec.c`

**Procedure name**
None.

---

**EZD0855I Option -option value length exceeds limit of limit characters**

**Explanation**
The option value specified exceeds the character limit.
**option** is the option that was specified.

**limit** is the maximum number of characters allowed for the option value.

**System action**
The **ipsec** command processing ends.

**Operator response**
See the information about managing network security in *z/OS Communications Server: IP System Administrator's Commands* or issue the **man ipsec** command in a z/OS UNIX shell to obtain information about the **ipsec** command syntax and options.

**System programmer response**
None.

**Module**
ipsec.c

**Procedure name**
None.

---

**EZD0856I**  **Option -opt does not support value val**

**Explanation**
The identified option requires a specific set of values and the specified value does not belong to that set.

**opt** is the option specified.

**val** is the unsupported value.

**System action**
The **ipsec** command processing ends.

**Operator response**
See the information about managing network security in *z/OS Communications Server: IP System Administrator's Commands* or issue the **man ipsec** command in a z/OS UNIX shell to obtain information about the **ipsec** command syntax and options.

**System programmer response**
None.

**Module**
ipsec.c

**Procedure name**
None.

---

**EZD0857I**  **Primary option not specified**

**Explanation**
The **ipsec** command requires a primary option but none was provided.
System action
The **ipsec** command processing ends.

Operator response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the **ipsec** command syntax and options.

System programmer response
None.

Module
ipsec.c

Procedure name
None.

**EZD0858I** Primary option `-opt1` does not support option `-opt2`

Explanation
The secondary option `opt2` is not allowed with the primary option `opt1`.

`opt1` is the primary option of the command.

`opt2` is the unsupported option that was used with `opt1`.

System action
The **ipsec** command processing ends.

Operator response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the **ipsec** command syntax and options.

System programmer response
None.

Module
ipsec.c

Procedure name
None.

**EZD0859I** Requester is not authorized to perform the request

Explanation
Different functions of the **ipsec** command require authorization with the system Access Control Facility. The requester does not have authority to perform the requested action.
System action
The `ipsec` command processing ends.

Operator response
This error is often caused by entering the wrong stack name with the `-p` option. Check the syntax, and if the problem persists, contact the system programmer to receive appropriate authority to use the `ipsec` command.

System programmer response
Give the user the appropriate authority to use the `ipsec` command. See the information about managing network security in z/OS Communications Server: IP System Administrator’s Commands for more information about the RACF® permission required by the `ipsec` command.

Module
ipsec.c

Procedure name
None.

EZD0860I Stack `stackname` is not available : `errno` `errno (description)` `errnojr` `errnojr`

Explanation
The identified stack could not be reached.
`stackname` is the stack that is unavailable.
`errno` is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.
`description` describes the meaning of `errno`.
`errnojr` is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The `ipsec` command processing ends.

Operator response
Ensure that the specified stack is valid and operational on the system.

System programmer response
None.

Module
ipsec.c

Procedure name
None.

EZD0861I Stack `stackname` is not configured for IPSECURITY
**Explanation**
The identified stack is known to the system, but is not configured for IPSECURITY. The `ipsec` command interacts only with IPSECURITY stacks.

`stackame` is the name of the stack that is not configured for IPSECURITY.

**System action**
The `ipsec` command processing ends.

**Operator response**
Contact the system programmer to determine whether the specified stack is intended to have IPSECURITY.

**System programmer response**
Check the profile for the specified stack to insure that IPSECURITY is configured correctly. See the z/OS Communications Server: IP Configuration Reference for more information about specifying IPSECURITY for a stack.

**Module**
`ipsec.c`

**Procedure name**
None.

**EZD0862I**  A system error kept the request from completing : `errno` `errno` (description) `errnojr` `errnojr`

**Explanation**
The system failed to complete a request due to a system error as indicated by the `errno` information provided.

`errno` is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

`description` describes the meaning of `errno`.

`errnojr` is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
The `ipsec` command processing ends.

**Operator response**
The error might be transient and reissuing the request might succeed. If the error persists, contact the system programmer.

**System programmer response**
Trace or log entries might give more information about the nature of the error. Use `errno`, `description`, and `errnojr` to fix the problem.

**Module**
`ipsec.c`
Procedure name
None.

EZD0863I  File system access error : errno errno (description) errnojr errnojr

Explanation
The request required a file system access that could not be completed.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

description describes the meaning of errno.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The **ipsec** command processing ends.

Operator response
The requester might not have the authority to perform the specified **ipsec** command action. Contact the system programmer to check authorization or about potential problems with the file system.

System programmer response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands for information about **ipsec** command requirements.

Module
ipsec.c

Procedure name
None.

EZD0864I  Memory could not be obtained to complete the request

Explanation
Memory could not be obtained to hold the system information that is being collected to satisfy the request.

System action
The **ipsec** command processing ends.

Operator response
The error might be transient and reissuing the request might succeed. If the error persists, contact the system programmer.

System programmer response
Trace or log entries might give more information about the nature of the error. Ensure that there is enough memory available on the system.
Module
ipsec.c

Procedure name
None.

EZD0865I  Could not set up signal handler : errno errno (description) errnojr errnojr

Explanation
An error occurred during the establishment of a signal handler in support of the command request.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errno) information in z/OS UNIX System Services Messages and Codes.

description describes the meaning of errno.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojr) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The ipsec command processing ends.

Operator response
Contact the system programmer.

System programmer response
Correct the error indicated by errno, errnojr, or description. It might also be helpful to reissue the command with the debug (-d) option.

Module
ipsec.c

Procedure name
None.

EZD0866I  Stack stackname is not available - but request is allowed to continue

Explanation
A reload/default filter function was requested for the specified stack, but this stack is not active on the system.

stackname is the name of the stack that is not available.

System action
The ipsec command is allowed to continue, but because the stack is unknown, the command might have no effect.

Operator response
Verify that the specified stack name is valid and reissue the command.

System programmer response
None.
**EZD0867I**  The `ipsec` command is not APF authorized

**Explanation**
The `ipsec` command is an APF-authorized application, but the version of the executable does not have the APF bit set.

**System action**
The `ipsec` command processing ends.

**Operator response**
Contact the system programmer.

**System programmer response**
Ensure that the `ipsec` command is installed correctly. Use the `extattr` command to ensure that the APF-authorized attribute is set to `on`.

**EZD0868I**  Name length exceeds limit of `limit` and is ignored - name starts with `partialname`

**Explanation**
A name was found to exceed the maximum length allowed.

`limit` is the maximum length allowed for the name.

`partialname` is the beginning of the string that exceeded the allowable length.

**System action**
The `ipsec` command continues using the other names in the command specification.

**Operator response**
Correct the name and reissue the command.

**System programmer response**
None.
EZD0869I  Invalid tunnel ID tunnelID is ignored

Explanation
The tunnel ID that was specified in the -a option is not in a valid format.

- The first character must be M, Y, or K, depending on the ipsec command that was issued:
  -m command
    A manual tunnel ID must be specified, and the first character must be M.
  -y command
    A dynamic tunnel ID must be specified, and the first character must be Y.
  -k command
    An IKE tunnel ID must be specified and the first character must be K.
  -f command
    Either a manual or dynamic tunnel ID can be specified and the first character can be either M or Y.
- The remaining characters must be numeric values in the range of 0–4294967295.

System action
Processing continues.

Operator response
Correct the tunnel ID and reissue the ipsec command.

System programmer response
None.

Module
ipsec.c

EZD0870I  Primary option -opt1 conflicts with primary option -opt2

Explanation
Only one primary option can be specified.

opt1 and opt2 are the primary option values that are in conflict.

System action
The ipsec command processing ends.

Operator response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the man ipsec command in a z/OS UNIX shell to obtain information about the ipsec command syntax and options.
**EZD0871I  All selection criteria were found in error**

**Explanation**
The `ipsec` command could not complete because all selection criteria entries were incorrect. Message EZD0869I is also issued for each name that is not valid.

**System action**
The `ipsec` command processing ends.

**Operator response**
Correct the specified names as indicated in the EZD0869I messages.

**System programmer response**
None.

---

**EZD0872I  Scope value conflicts with selection criteria**

**Explanation**
The command as specified conflicts with the indicated scope value. 
value is the specified scope and is either `current`, `policy`, or `profile`.

**System action**
The `ipsec` command processing ends.

**Operator response**
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

**System programmer response**
None.
**EZD0873I** Function *function* conflicts with option -*option*

**Explanation**
The function that the *ipsec* command is attempting to perform does not allow the specified option.

**System action**
The *ipsec* command processing ends.

**Operator response**
See the information about managing network security in *z/OS Communications Server: IP System Administrator's Commands* or issue the `man ipsec` command in a *z/OS UNIX* shell to obtain information about the *ipsec* command syntax and options.

**System programmer response**
None.

---

**EZD0875I** Selection criteria option *opt1* conflicts with selection criteria option *opt2*

**Explanation**
On any command request, only one type of selection criteria option can be specified.

*opt1* and *opt2* are the selection criteria option values in conflict.

**System action**
The *ipsec* command processing ends.

**Operator response**
See the information about managing network security in *z/OS Communications Server: IP System Administrator's Commands* or issue the `man ipsec` command in a *z/OS UNIX* shell to obtain information about the *ipsec* command syntax and options.

**System programmer response**
None.
Procedure name
None.

EZD0876I  Invalid type data specified

Explanation
The IP Traffic Test option for the `ipsec -t` command contains an incorrect value.

$type$ is the type of option that is incorrect. Possible values are:

- SourceIpAddress
- DestIpAddress
- Protocol
- SourcePort
- DestinationPort
- Direction
- SecClass

$data$ is the incorrect data that was specified.

System action
The `ipsec` command processing ends.

Operator response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands for information about the correct values for the IP Traffic Test option of the `ipsec` command. Correct the IP Traffic Test option and reissue the command.

System programmer response
None.

Module
ipsec.c

Procedure name
do_traflictest

EZD0877I  subfield required for specified option

Explanation
The IP Traffic Test option that was identified in the message for the `ipsec -t` command requires additional subfield data.

$subfield$ is the name of the field that is missing data.

$option$ is the IP Traffic Test option that requires the additional data.

System action
The `ipsec` command processing ends.
Operator response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands for information about the correct syntax for the IP Traffic Test option of the \texttt{ipsec} command. Correct the IP Traffic Test option and reissue the command.

System programmer response
None.

Module
ipsec.c

Procedure name
do Traffictest

\texttt{EZD0878I} Excessive -t argument \texttt{arg} encountered

Explanation
Following -t, the traffic test command takes a number of positional arguments as indicated by the syntax diagram. This message is issued because the specification of positional arguments does not follow the requirements of the syntax.

\texttt{arg} is the excessive data.

System action
The \texttt{ipsec} command processing ends.

Operator response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands for information about the correct syntax for the IP Traffic Test option of the \texttt{ipsec} command. Correct the IP Traffic Test option and reissue the command.

System programmer response
None.

Module
ipsec.c

Procedure name
None.

\texttt{EZD0879I} ipsec function cannot be issued while default policy is active

Explanation
The requested ipsec function cannot be issued against default policy.

\texttt{function} is the requested ipsec function.

System action
The \texttt{ipsec} command processing ends.
Operator response
If this command is required, see the information about managing network security in z/OS Communications
Server: IP System Administrator's Commands or issue the man ipsec command in a z/OS UNIX shell for
information about switching from the default policy.

System programmer response
None.

Module
ipsec.c

Procedure name
do_manual

EZD0880I  No filters exist under the specified scope

Explanation
No filters exist for the requested scope.

System action
The ipsec command processing ends.

Operator response
None.

System programmer response
None.

Module
ipsec.c

Procedure name
do_filters

EZD0881I  Connect error for IKE daemon connection : errno errno (description) errnojr errnojr

Explanation
An error occurred during a connect to the IKE daemon.
errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return
codes (ernos) information in z/OS UNIX System Services Messages and Codes.
description describes the meaning of errno.
errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is
explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services
Messages and Codes, where the reason codes are listed.

System action
The ipsec command processing ends.
Operator response
Ensure that the IKE daemon is running and use errno, description, and errnojr to fix the problem.

System programmer response
None.

Module
ipsec.c

Procedure name
None.

EZD0882I Write error on IKE daemon connection: errno errno (description) errnojr errnojr

Explanation
An error occurred during a write operation to the IKE daemon.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

description describes the meaning of errno.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The ipsec command processing ends.

Operator response
Ensure that the IKE daemon is running and use errno, description, and errnojr to fix the problem.

System programmer response
None.

Module
ipsec.c

Procedure name
request_iked

EZD0883I Read error on IKE daemon connection: errno errno (description) errnojr errnojr

Explanation
An error occurred during a read operation to the IKE daemon.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

description describes the meaning of errno.
errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
The `ipsec` command processing ends.

**Operator response**
Ensure that the IKE daemon is running and use `errno`, `description`, and `errnojr` to fix the problem.

**System programmer response**
None.

**Module**
`ipsec.c`

**Procedure name**
`request_iked`

---

**EZD0884I**  Illegal data received over IKE daemon connection

**Explanation**
The data that was read over the IKE daemon connection was not in the expected format.

**System action**
The `ipsec` command processing ends.

**Operator response**
Data might have been corrupted in transmission. Try the command again.

**System programmer response**
None.

**Module**
`ipsec.c`

**Procedure name**
`request_iked`

---

**EZD0885I**  Options `-opt1` and `-opt2` cannot be specified together

**Explanation**
The `ipsec` command was issued with the two indicated options. These options cannot be specified together. `opt1` and `opt2` are the names of the options that were incorrectly specified together.

**System action**
The `ipsec` command processing ends.
Operator response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

System programmer response
None.

Module
ipsec.c

Procedure name
validity_check

EZD0886I  Function `function` conflicts with option `-opt optvalue`

Explanation
The specified option value is not valid for the function specified.
`function` is the primary function the `ipsec` command was attempting to perform.
`opt` is the option that was specified.
`optvalue` is the option value that was not valid.

System action
The `ipsec` command processing ends.

Operator response
See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

System programmer response
None.

Module
ipsec.c

Procedure name
validity_check

EZD0887I  IKE daemon could not process the request due to an internal error

Explanation
A request was made to the IKE daemon that could not be completed.

System action
The `ipsec` command processing ends.
Operator response
Contact the system programmer.

System programmer response
Contact IBM software support services and provide a syslog that covers the time of the command failure. If available, provide a CTRACE for component SYSTCPIK.

Module
ipsec.c

Procedure name
None.

EZD0888I  IKE daemon could not process the request - stack stackname is unknown to IKE daemon

Explanation
The command specifies a stack that cannot be verified by the IKE daemon.

stackname is the name of the stack that cannot be verified.

System action
The ipsec command processing ends.

Operator response
Ensure that the stack name as specified is correct in the current system configuration. From the operator console, use the d tcpip command to list all available stacks.

System programmer response
None.

Module
ipsec.c

Procedure name
None.

EZD0889I  Could not create IKE daemon home directory path : errno erro (description) errojr errojr

Explanation
Users of the ipsec command must have the authority to access the IKE daemon home directory.

path is the path of the IKE daemon home directory.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

description describes the meaning of errno.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.
System action
The \texttt{ipsec} command processing ends.

Operator response
Contact the system programmer.

System programmer response
See the information about managing network security in \textit{z/OS Communications Server: IP System Administrator’s Commands} or issue the \texttt{man ipsec} command in a \texttt{z/OS UNIX} shell to obtain information about the \texttt{ipsec} command syntax and group access control requirements.

Module
ipsec.c

Procedure name
None.

\textbf{EZD0890I} \hspace{1em} \texttt{Could not open code page converters : errno \textit{errno} \textit{description} errnojr \textit{errnojr}}

Explanation
Command execution requires a code page converter of IBM-1047 (EBCDIC) and a code page converter for the local code page.

\textit{errno} is the \texttt{z/OS UNIX} System Services return code. These return codes are listed and described in the \textit{return codes (erros)} information in \textit{z/OS UNIX System Services Messages and Codes}.

\textit{description} describes the meaning of \textit{errno}.

\textit{errnojr} is the hexadecimal \texttt{z/OS UNIX} System Services reason code. The format of the 4-byte reason code is explained in the introduction to the \textit{reason codes (errnojrs)} information in the \textit{z/OS UNIX System Services Messages and Codes}, where the reason codes are listed.

System action
The \texttt{ipsec} command processing ends.

Operator response
Contact the system programmer.

System programmer response
Verify that code page converters are available for IBM-1047 and for the local code page. See the information about code page conversion in \textit{z/OS UNIX System Services User’s Guide} for more information about specifying code page converters.

Module
ipsec.c

Procedure name
None.

\textbf{EZD0891I} \hspace{1em} \texttt{Code page conversion failed and is ignored for name}
Explanation
The `ipsec` command encountered a name that could not be converted.

=name is the name that could not be converted.

System action
The `ipsec` command processing continues.

Operator response
Contact the system programmer.

System programmer response
See the information about code page conversion in z/OS UNIX System Services User's Guide for more information about specifying code page converters.

Module
ipsec.c

Procedure name
None.

EZD0892I  Signal `signum` received - command ends

Explanation
A system signal was received and forced the `ipsec` command to end. See z/OS UNIX System Services Command Reference for more information about signals.

`signum` identifies the signal that was received.

System action
The `ipsec` command processing ends.

Operator response
The condition might be temporary. Try the command again. It might be helpful to specify the debug (-d) option on the command specification. If the failure persists, contact the system programmer.

System programmer response
Contact IBM software support services and provide a syslog that covers the time of the command failure. If available, provide a CTRACE for component SYSTCPIK.

Module
ipsec.c

Procedure name
None.

EZD0893I  -`opt1` option can only be specified with the -`opt2` option

Explanation
The `ipsec` command was issued with -`opt2` specified, but without the required -`opt2` option specified.
opt1 is the `ipsec` command option that was specified.

opt2 is the required `ipsec` command option that is missing.

**System action**
The `ipsec` command processing ends.

**Operator response**
Reissue the `ipsec` command with both `-opt1` and `-opt2` specified. See the information about managing network security in `z/OS Communications Server: IP System Administrator's Commands` or issue the `man ipsec` command in a `z/OS UNIX` shell to obtain information about the `ipsec` command syntax and options.

**System programmer response**
None.

**Module**
ipsec.c

**Procedure name**
validity_check

**EZD0894I** More than one value specified for the `-opt` option

**Explanation**
Only one value can be specified for the indicated `ipsec` command option.

opt is the `ipsec` command option that was specified.

**System action**
The `ipsec` command processing ends.

**Operator response**
Remove extraneous values and reissue the `ipsec` command. See the information about managing network security in `z/OS Communications Server: IP System Administrator's Commands` or issue the `man ipsec` command in a `z/OS UNIX` shell to obtain information about the `ipsec` command syntax and options.

**System programmer response**
None.

**Module**
ipsec.c

**Procedure name**
parse_args

**EZD0895I** Incorrect `-opt` option value value is ignored

**Explanation**
An incorrect option value was specified and ignored.

opt is the `ipsec` command option that was specified.
value is the value that was specified for the ipsec option.

**System action**
The `ipsec` command continues using other specified values, if any, or the default value, if any.

**Operator response**
Specify an option value in the accepted value range and reissue the `ipsec` command. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

**System programmer response**
None.

**Module**
`ipsec.c`

**Procedure name**
`parse_args`

**EZD0896I** An IPv6 address was specified with a stack stackname that was not enabled for IPSECURITY

**Explanation**
The request specified an IPv6 address associated with the stack specified by the `stackname` value, but this stack does not have IPSECURITY defined in its IPCONFIG6 statement.

**System action**
The `ipsec` command processing ends.

**Operator response**
Contact the system programmer to determine whether the specified stack is intended to have IP security for its IPv6 configuration.

**System programmer response**
Check the profile for the specified stack to insure that IP security is configured correctly on the IPCONFIG6 statement. See the information about the IPCONFIG6 statement in the z/OS Communications Server: IP Configuration Reference for more information about specifying IP security for a stack.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP other application

**Module**
`ipsec.c`
**EZD0897I  Two addresses were specified that are not in the same address family**

**Explanation**
IP security is supported between two endpoints when both endpoints are IPv4 or both endpoints are IPv6. IP security is not supported between endpoints of different address families.

**System action**
The `ipsec` command processing ends.

**Operator response**
Issue the `ipsec` command request with two IPv4 addresses or two IPv6 addresses.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP

**Module**
ipsecc

---

**EZD0898I  IKE daemon could not process the request due to a Security Association negotiation failure**

**Explanation**
A refresh or activation request could not be completed because of an error with the Security Association negotiation. The error might be a problem with the configuration or an internal error with the IKE daemon.

**System action**
The `ipsec` command processing ends.

**System programmer response**
Look at the syslog daemon syslog file that covers the time of the command failure. The negotiation failure might be caused by a configuration problem or by an internal error. Messages that see policy might indicate that the request failed because of the policy configuration. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy. If the failure is caused by an IKE daemon internal error, contact IBM software support services and provide a syslog daemon syslog file that covers the time of the command failure. If available, provide a CTRACE for component SYSTCPIK. See the information about diagnosing IKE daemon problems in z/OS Communications Server: IP Diagnosis Guide for more information about CTRACE.

**User response**
Contact the system programmer.
Module
ipsec.c

Procedure name
None.

EZD0902I  Peer certificate failed validation - System SSL CMS error : \textit{gsk\_rc description}

Explanation
A call to the System SSL Certificate Management Services (CMS) API to validate the peer certificate returned an error.

\textit{gsk\_rc} is the hexadecimal CMS status code. See the information about the CMS status codes in \textit{z/OS Cryptographic Services System SSL Programming}. 

\textit{description} describes the meaning of \textit{gsk\_rc}.

System action
The operation being performed fails; IKE daemon processing continues.

Operator response
Use \textit{gsk\_rc} and \textit{description} to correct the error.

System programmer response
None.

Module
pki390.cpp

Procedure name
None.

EZD0903I  Peer certificate failed authentication - System SSL CMS error : \textit{gsk\_rc description}

Explanation
A call to the System SSL Certificate Management Services (CMS) API to authenticate the peer certificate returned an error.

\textit{gsk\_rc} is the hexadecimal CMS status code. See the information about the CMS status codes in \textit{z/OS Cryptographic Services System SSL Programming}. 

\textit{description} describes the meaning of \textit{gsk\_rc}.

System action
The operation being performed fails; IKE daemon processing continues.

Operator response
Use \textit{gsk\_rc} and the description provided to fix the error.

System programmer response
None.
EZD0904I  IKE CONFIGURATION FILE fname COULD NOT BE OPENED OR FILE DOES NOT EXIST

Explanation
The Internet Key Exchange (IKE) daemon could not open the file name that was specified for IKE configuration. fname is the name of the file that the IKE daemon could not open.

System action
If an error occurs during IKE startup, the IKE daemon configuration file processing is ended and the IKE daemon ends. If the error occurs due to a MODIFY REFRESH command, IKE daemon configuration file processing is ended, but the IKE daemon remains active. IKE configuration retains all values prior to the MODIFY REFRESH command.

Operator response
Verify that the file name provided is correct and that the file exists. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IkeConfig statement and IKE daemon configuration file.

System programmer response
None.

EZD0905I  IKE configuration file fname does not contain an IkeConfig statement

Explanation
The Internet Key Exchange (IKE) daemon configuration file must specify an IkeConfig statement to indicate that there are IKE configuration parameters. fname is the name of the IKE daemon configuration file.

System action
If the error occurs during IKE startup, IKE daemon configuration file processing ends, and the IKE daemon ends. If the error occurs due to a MODIFY REFRESH command, IKE daemon configuration file processing ends, but the IKE daemon remains active. IKE configuration retains all values prior to the MODIFY REFRESH command.

Operator response
Ensure that IkeConfig statement is specified in the configuration file. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IkeConfig statement and IKE daemon configuration file.
System programmer response
None.

Module
ike_config.cpp

Procedure name
None.

EZD0906I  Unknown IKE configuration file parameter pname on line linenum

Explanation
The Internet Key Exchange (IKE) daemon configuration file contains a parameter that is not recognized by the IKE daemon.

In the message text:

pname
   The unknown parameter.

linenum
   The line of the configuration file where the parameter was found.

System action
If the error occurred during IKE startup, IKE daemon configuration file processing ends, and the IKE daemon ends. If the error occurred as the result of a MODIFY REFRESH command, the IKE daemon configuration file processing ends, but the IKE daemon remains active. IKE configuration retains all values prior to the MODIFY REFRESH command.

Operator response
Contact the system programmer.

System programmer response
Change the parameter specified by the pname value to a valid IKE parameter or remove that parameter. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp

Routing code
10
Example
EZD0906I Unknown IKE configuration file parameter junk on line 17

EZD0907I  Missing value for IKE configuration file parameter pname on line linenum

Explanation
The Internet Key Exchange (IKE) daemon configuration file contains a parameter *pname* that requires a parameter value and no value was specified.

In the message text:

- **pname**
  - The parameter name that is missing a value.

- **linenum**
  - The line of the configuration file where the parameter was found.

System action
If the error occurs during IKE startup, IKE daemon configuration file processing ends, and the IKE daemon ends. If the error occurs as a result of a MODIFY REFRESH command, IKE daemon configuration file processing ends, but the IKE daemon remains active. IKE configuration retains all values prior to the MODIFY REFRESH command.

Operator response
Contact the system programmer.

System programmer response
Provide a value for the parameter specified by the *pname* value. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file and valid parameters for the parameter specified by the *pname* value.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp

Routing code
10
Example
EZD0907I Missing value for IKE configuration file parameter NssWaitRetries on line 7

EZD0908I IKE configuration file parameter *pname* does not support value *val* on line *linenum*

Explanation
The Internet Key Exchange (IKE) daemon configuration file contains a parameter *pname* that contains an unsupported value *val*.

In the message text:

*pname*
The parameter name.

*val*
The unsupported parameter value.

*linenum*
The line of the configuration file where the parameter was found.

System action
If the error occurs during IKE startup, IKE daemon configuration file processing ends, and the IKE daemon ends.
If the error occurs as a result of a MODIFY REFRESH command, IKE daemon configuration file processing ends, but the IKE daemon remains active. IKE configuration retains all values prior to the MODIFY REFRESH command.

Operator response
Contact the system programmer.

System programmer response
Insert a supported value for the parameter specified by the *pname* value. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file and supported values for the parameter specified by the *pname* value.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp

Routing code
10
Descriptor code
12

Example
EZD0908I IKE configuration file parameter NssWaitLimit does not support value 301 on line 7

EZD0909I INCORRECT SYNTAX ON MODIFY COMMAND OPTION opt

Explanation
The MODIFY command that was entered could not be parsed because of syntax problems. The failure occurred while parsing opt.

opt is the command option that was specified with the MODIFY command that is incorrect.

System action
The MODIFY command is rejected. IKE daemon configuration retains all values prior to the MODIFY REFRESH command.

Operator response
See the information about the MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information the correct syntax of the MODIFY command.

System programmer response
None.

Module
mdfysrvr.c

Procedure name
None.

EZD0910I IKE configuration file was not specified - using defaults for all IKE configuration parameters

Explanation
No Internet Key Exchange (IKE) daemon configuration file was found.

System action
The IKE daemon will use default values for all IKE daemon configuration parameters; the IKE daemon continues.

Operator response
If the intention was to use an IKE daemon configuration file, see the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file.

System programmer response
None.

Module
ike_config.cpp
EZD0911I   IKE CONFIG PROCESSING COMPLETE USING FILE fname

Explanation
The Internet Key Exchange (IKE) daemon configuration file processing completed using file fname. fname is the file name of the IKE daemon configuration file.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

Module
ike_config.cpp

Procedure name
None.

EZD0912I   IKE configuration file contains more than one instance of non-repeatable parameter parm-value of last specified parameter will be used

Explanation
The Internet Key Exchange (IKE) daemon configuration file contains more than one instance of a non-repeatable parameter parm.
parm is the non-repeatable parameter that was specified more than once.

System action
The value of the last specified parameter will be used; IKE daemon processing continues.

Operator response
If this message was unexpected, see the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file and which parameters are repeatable or non-repeatable.

System programmer response
None.

Module
ike_config.cpp

Procedure name
None.
EZD0913I   IKE configuration file parsing error - explanation

Explanation
The Internet Key Exchange (IKE) daemon encountered an error while parsing the IKE daemon configuration file. Possible errors are:

• Unexpected end of file (EOF) encountered.
• Syntax errors in the IKE daemon configuration file.

explanation describes the IKE daemon configuration file parsing error.

System action
If the error occurs during IKE daemon startup, IKE daemon configuration file processing is ends, and the IKE daemon ends. If the error occurs due to a MODIFY REFRESH command, IKE daemon configuration file processing ends, but the IKE daemon remains active. IKE daemon configuration retains all values prior to the MODIFY REFRESH command.

Operator response
Use explanation to fix the problem. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file.

System programmer response
None.

Module
ike_config.cpp

Procedure name
None.

EZD0915I   IKE DAEMON INITIALIZED WITH WARNINGS

Explanation
The Internet Key Exchange (IKE) daemon initialized with non-fatal errors.

System action
IKE daemon processing continues.

Operator response
Examine the syslog for errors that occurred during the IKE daemon initialization that has the same message instance number as this message for more information. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System programmer response
None.

Module
main.cpp.
**Procedure name**

None.

**EZD0917I  Could not find applicable KeyExchangeRule - LocalIp : LSIP RemoteIp : RSIP LocalID : LSID RemoteID : RSID**

**Explanation**

The Internet Key Exchange (IKE) daemon could not find an applicable KeyExchangeRule statement for the specified classification. The classification consists of the local security endpoint IP address (LSIP), remote security endpoint IP address (RSIP), local security endpoint identity (LSID), and remote security endpoint identity (RSID). If the remote system is behind a NAT, ensure that the RemoteSecurityEndpoint location in the KeyExchangeRule is the public address of the remote system.

**System action**

The Security Association (SA) activation failed; IKE daemon processing continues.

**Operator response**

Add a suitable KeyExchangeRule statement for the classification to the IPSec policy, if necessary. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**System programmer response**

None.

**Module**

policymgr.cpp

**EZD0918I  ICSF service CSNBSYE failed for AES encryption : return code = rc reason code = rsn**

**Explanation**

The Integrated Cryptographic Service Facility (ICSF) service returned an error when called to perform AES encryption.

In the message text:

- \( rc \)
  - The hexadecimal return code returned from the ICSF function call.

- \( rsn \)
  - The hexadecimal reason code returned from the ICSF function call.

**System action**

The operation being performed fails; IKE daemon processing continues.

**Operator response**

Verify that ICSF is running. See the information about the ICSF and TSS return and reason Codes information in z/OS Cryptographic Services ICSF Application Programmer’s Guide for the meaning of the \( rc \) and \( rsn \) values and for specific actions to be taken.
System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Module
aes_obj.cpp

EZD0919I Local security endpoint IP address cannot be determined from IP address range (ip_range)
errid = errid

Explanation
While attempting to activate a dynamic tunnel, the IKE daemon could not determine the local security endpoint IP address for the tunnel. Usually this is caused by specifying IP addresses and the RemoteIpGranularity and LocalIpGranularity parameters on the IpLocalStartAction statement in the configuration policy that resolve to ranges of IP addresses.

ip_range is the address range from which the IKE daemon was trying to obtain the endpoint location.

errid is a unique error ID for the occurrence of this message.

System action
The dynamic tunnel activation failed; IKE daemon processing continues.

Operator response
Review and alter the IPSec configuration as necessary in order for the IKE daemon to be able to determine a local IP address. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy. After you review your configuration, if you believe that the IKE daemon should be able to determine the source IP address, contact the system programmer.

System programmer response
Contact IBM software support services with the errid and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

Module
policymgr.cpp

Procedure name
PMGet_Phase1Rule

EZD0920I Remote security endpoint IP address cannot be determined from IP address range (ip_range)
(errid = errid)

Explanation
While attempting to activate a dynamic tunnel, the IKE daemon failed to determine the remote security endpoint IP address for the tunnel.
ip_range is the address range from which the IKE daemon was trying to obtain the endpoint location. errid is a unique error ID for the occurrence of this message.

**System action**
The dynamic tunnel activation failed; IKE daemon processing continues.

**Operator response**
Review and alter the IPSec configuration as necessary. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy. After you review your configuration, if you believe that the IKE daemon should be able to determine the source IP address, contact the system programmer.

**System programmer response**
Contact IBM software support services with the errid and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

**Module**
policymgr.cpp

**Procedure name**
PMGet_Phase1Rule

**EZD0922I INTERNAL ERROR errid - value1 | value2 | value3**

**Explanation**
The Internet Key Exchange (IKE) daemon detected an internal error. Additional diagnostic messages that have the same message instance number might be issued. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

errid is a unique ID for this error.
value1 is optional error information.
value2 is optional error information.
value3 is optional error information.

**System action**
Results are unpredictable. One or more address space dumps are produced with dump titles that match the message text.

**Operator response**
Contact the system programmer.

**System programmer response**
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK. If available, provide any dumps associated with this message.

**Module**
Numerous.
Procedure name
None.

**EZD0923I** IKE HAS RECEIVED THE STOP COMMAND

Explanation
The Internet Key Exchange (IKE) daemon received the STOP command.

System action
The IKE daemon ends.

Operator response
None.

System programmer response
None.

Module
mdfysrvr.c

Procedure name
None.

**EZD0924I** Transform transform_id : unable to store Diffie-Hellman group ( DH_group_id ) reason
(reason)

Explanation
An IKE negotiation failed because the server was unable to store the specified Diffie-Hellman (DH) group identifier during the verification of the transforms in a proposal. This was caused by an unsupported protocol or group descriptor.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

transform_id is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

DH_group_id is the DH group ID. Only groups 1, 2, 5, and 14 are supported.

reason is either protocol or descriptor, which indicates whether the error was with the protocol or the group descriptor.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Request that the administrator of the remote security endpoint verify that only valid protocols and DH groups were specified.
**EZD0925I**   Transform \textit{transform\_id} : unsupported life-type attribute (\textit{lifetype\_id})

**Explanation**
An IKE negotiation failed because the value specified for the life-type attribute is not in seconds or kilobytes.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

\textit{transform\_id} is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

\textit{lifetype\_id} is the ID type of the unsupported life-type attribute.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Request that the administrator of the remote security endpoint ensure that their life-type values are specified as seconds or kilobytes.

**System programmer response**
None.

**EZD0926I**   Transform \textit{transform\_id} : key length specified for fixed key length algorithm or the algorithm is not supported

**Explanation**
An IKE negotiation failed because a key length is specified for an algorithm that has a fixed key length or the algorithm is unsupported.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

\textit{transform\_id} is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference.
The SA negotiation failed; IKE daemon processing continues.

Operator response
Request that the administrator of the remote security endpoint verify that their transform algorithms are specified correctly.

System programmer response
None.

Module
ah_lf.cpp, esp_lf.cpp, oakley_lf.cpp

Procedure name
None.

EZD0927I Transform transform_id : life value not specified for life-type attribute

Explanation
An IKE negotiation failed because a transform specified a life-type attribute with no corresponding life value.
Additional diagnostic messages with the same message instance number will be issued to identify the impacted security association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

transform_id is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Request that the administrator of the remote security endpoint verify that they specified life values for all life-type attributes.

System programmer response
None.

Module
ah_lf.cpp, esp_lf.cpp, gen.cpp, oakley_lf.cpp

Procedure name
None.

EZD0928I Transform transform_id : unsupported attribute tag ( attribute_tag )
**Explanation**

An IKE negotiation failed because a transform specified an attribute that is not known for the specified transform.

Additional diagnostic messages with the same message instance number will be issued to identify the impacted security association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*transform_id* is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

*attribute_tag* is the ID that represents the unknown attribute.

**System action**

The SA negotiation failed; IKE daemon processing continues.

**Operator response**

Request that the administrator of the remote security endpoint verify that all the transforms are specified correctly.

**System programmer response**

None.

**Module**

ah_lf.cpp, esp_lf.cpp, gen.cpp, oakley_lf.cpp

**Procedure name**

None.

**EZD0929I**  
Transform *transform_id*: attribute length ( *attr_length* ) incorrect size for attribute ( *attribute* )

**Explanation**

An IKE negotiation failed because a transform specified an attribute length that is the incorrect size for the specified attribute.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*transform_id* is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

*attr_length* is the length of this attribute in bytes. 

*attribute* is a string representation of the attribute. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about transform attributes.

**System action**

The SA negotiation failed; IKE daemon processing continues.
Operator response
Request that the administrator of the remote security endpoint verify that their transforms are specified correctly.

System programmer response
None.

Module
ah_if.cpp, esp_if.cpp, oakley_if.cpp, gen.cpp

Procedure name
None.

EZD0930I  Transform transform_id: unable to read attribute

Explanation
An IKE negotiation failed because the length calculated for the next attribute is larger than the remaining attribute string.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

transform_id is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Request that the administrator of the remote security endpoint verify that their transforms are configured correctly.

System programmer response
None.

Module
ah_if.cpp, esp_if.cpp, oakley_if.cpp

Procedure name
None.

EZD0931I  Transform transform_id: no encapsulation mode specified

Explanation
An IKE negotiation failed because the specified transform does not specify tunnel or transport mode.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.
**transform_id** is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in *Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference*. Phase 1 transforms are specified on a `KeyExchangeOffer` statement, and phase 2 transforms are specified on an `IpDataOffer` statement. Phase 2 transforms specify the encapsulation mode on the `HowToEncap` statement.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Request that the administrator of the remote security endpoint verify that they specified tunnel or transport mode for this negotiation.

**System programmer response**
None.

**Module**
`ah lf.cpp, esp lf.cpp`

**Procedure name**
None.

**EZD0932I Transform transform_id: no authentication algorithm specified**

**Explanation**
An IKE negotiation failed because the specified transform does not specify an authentication algorithm.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

`transform_id` is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in *Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference*. Phase 1 transforms are specified on a `KeyExchangeOffer` statement, and phase 2 transforms are specified on an `IpDataOffer` statement.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Request that the administrator of the remote security endpoint ensure that an authentication algorithm is specified for each transform.

**System programmer response**
None.

**Module**
`gen.cpp, ah lf.cpp`

**Procedure name**
None.
EZD0933I  Transform transform_id : no hash algorithm specified

Explanation
An IKE negotiation failed because the specified transform does not specify a hash algorithm.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

transform_id is the value that is used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Request that the administrator of the remote security endpoint ensure that a hash algorithm is specified for each transform.

System programmer response
None.

Module
gen.cpp, oakley_lf.cpp

Procedure name
None.

EZD0934I  Unsupported exchange type (xchg_id) for phase 1 security association

Explanation
An IKE negotiation failed because the IKE daemon encountered an unsupported exchange type.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted security association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

xchg_id is the number that identifies the exchange type that was not valid.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to ensure that a supported exchange type is used in negotiations.
Module
policy.cpp

Procedure name
None.

EZD0935I  No local policy data available

Explanation
An IKE negotiation failed because the IKE daemon found no suites for matching proposals.
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Check the local policy and ensure that it exists and is valid. Local policies are specified on KeyExchangeAction and IpDataOffer Statements. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

System programmer response
None.

Module
policy.cpp

Procedure name
None.

EZD0936I  Phase 2 IDs must be IP type addresses

Explanation
An IKE negotiation failed because the identities exchanged during a phase 2 Security Association (SA) negotiation were not IP-type addresses.
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to ensure that phase 2 IDs are IP-type addresses.
**Module**  
oakley_phaseII.cpp, policy.cpp

**Procedure name**  
None.

**EZD0937I** Could not add certificate with label (label) to the supported certificate authority list

**Explanation**  
The IKE server could not find a certificate with the specified label on the key ring identified by the Key ring setting for the z/OS Image. This label was defined to be in the specified Key Ring database. In the IKE daemon configuration file, this label corresponds to the SupportedCertAuth parameter of the IkeConfig statement.  
*label* is the label of this certificate.

**System action**  
The IKE server will ignore the label; IKE daemon processing continues.

**Operator response**  
Verify that the label is correct.

When configured without the IBM Configuration Assistant for z/OS Communications Server:
- Verify that the label specified on the SupportedCertAuth parameter of the IkeConfig statement is correct.
- Verify that the key ring identified by the KeyRing parameter of the IkeConfig statement is correct.

See the information about the IKE daemon in *z/OS Communications Server: IP Configuration Reference* for more information about the IkeConfig statement.

When configured with the IBM Configuration Assistant for z/OS Communications Server:
- Verify that a certificate with that label is connected to the key ring identified by the Key ring database configured on the IPSec: IKE Daemon Settings panel.
- Verify that the key ring database name and location are configured correctly.

See the online helps in the GUI for additional information.

**System programmer response**  
None.

**Module**  
cacache.cpp

**Procedure name**  
None.

**EZD0938I** Unsupported algorithm (algorithm) found in transform

**Explanation**  
The IKE negotiation failed because the algorithm specified in a transform is not supported.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:
**Algorithm**
The numerical value of the unsupported algorithm. Supported algorithms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 algorithms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement. Go to the http://www.iana.org/assignments/isakmp-registry website for additional information about algorithm number assignments.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Notify the administrator of the remote security endpoint about this error and ask the administrator to ensure that all hash, encryption, and authentication algorithms are specified correctly for all transforms.

**Module**
ipsecklen.cpp, oakley_mf.cpp

**Procedure name**
None.

**EZD0939I  Bad proposal layout for protocol (protocol_id)**

**Explanation**
An IKE negotiation failed because the proposal layout for the specified protocol is not valid. Possible reasons for the error are too many suites in an offer or a proposal that is too long or too short.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

protocol_id is the ID of the protocol.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Notify the administrator of the remote security endpoint that you received a proposal that was not valid and ask the administrator to ensure that their policy is configured correctly.

**Module**
policy.cpp

**Procedure name**
None.
EZD0940I Wrong security parameter index(SPI) size (SPI_size_exp / SPI_size) for protocol (protocol_id) in proposal (proposal)

Explanation
An IKE message failed to be processed because an SPI size was not valid.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

SPI_size_exp is the expected SPI size.
SPI_size is the SPI size found.
protocol_id is the protocol ID.
proposal is the proposal number or -1 if this message was not part of an SA negotiation.

System action
Contact the system programmer.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received a proposal that was not valid and ask the administrator to ensure that their policy is configured correctly.

Module
doi.cpp, infoXchg.cpp, policy.cpp

Procedure name
None.

EZD0941I More than 1 object in proposal reply for protocol (protocol_id)

Explanation
An IKE negotiation failed because more than one transform was specified in a reply or more than one proposal was chosen on a match of an offered proposal.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

object is either suite or transform. If object is suite, then more than one proposal was chosen by the remote security endpoint. If object is transform, then more than one transform was specified for a proposal.

protocol_id is the protocol ID.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.
System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to ensure that only one proposal is accepted from an offer and only one transform is specified in a proposal.

Module
policy.cpp

Procedure name
None.

EZD0942I Duplicate protocol ( protocol ) found in proposal ( proposal_num )

Explanation
An IKE daemon negotiation failed because the reply proposal contains more than one transform with the specified protocol.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

protocol is the protocol ID.

proposal_num is the proposal number.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to ensure that they do not specify the same protocol more than once in a single proposal.

Module
policy.cpp

Procedure name
None.

EZD0943I Proposal ( proposal_num ) contains too few bytes ( bytes )

Explanation
An IKE negotiation failed because the proposal does not contain enough data.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

proposal_num is the proposal number.

bytes is the number of bytes missing from the proposal.
System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received a proposal that was not valid and ask the administrator to ensure that their policy is configured correctly.

Module
policy.cpp

Procedure name
None.

EZD0944I Transform transform_id : duplicate attribute (attribute)

Explanation
An IKE negotiation failed because a transform specified the same attribute more than the allowed number of times.

Additional diagnostic messages with the same message instance number will be issued to identify the impacted security association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

transform_id is the value used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

attribute is the attribute that was duplicated. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about transform attributes.

System action
This SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to ensure that all attributes are specified correctly. With the exception of the life-value tag, attributes can be specified only once in a single transform.

Module
gen.cpp

Procedure name
None.

EZD0945I No proposal specified or proposal is empty
Explanation
An IKE negotiation failed because a proposal does not exist or contains no data.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received a proposal that was not valid and ask the administrator to ensure that their policy is configured correctly.

Module
policy.cpp

Procedure name
None.

EZD0946I Protocol mismatch: IpDataOffer (offer_num) requires (reqd_proto) but proposal (prop_num) includes (prop_proto)

Explanation
An IKE phase 2 negotiation encountered a protocol mismatch.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

offer_num is the number of an IpDataOffer referenced from an IpDynVpnAction. The number corresponds to the order of the references in the IpDynVpnAction. Therefore, the first IpDynVpnOffer referenced from the IpDynVpnAction would have number 1 in this message.

reqd_proto is the description of the protocols configured in the IpDataOffer indicated by the offer_num value. Possible values include AH, ESP, or AH+ESP. AH+ESP indicates that the combination of AH and ESP are required.

prop_num is the proposal number from the remote security endpoint.

prop_proto is the description of the protocols proposed in the proposal indicated by prop_num. Possible values include AH, ESP, or AH+ESP. AH+ESP indicates that the combination of AH and ESP was proposed.

System action
The IKE negotiation might succeed if a different proposal is found and accepted. If an acceptable proposal is not found, the IKE negotiation fails. If the negotiation fails, message EZD1022I will be issued, which will identify the IpDynVpnAction that referenced the IpDataOffer indicated by the offer_num value. IKE daemon processing continues.

Operator response
If the proposal that contains the mismatch is the one that should be accepted, take one of the following actions:
• Alter the local policy to accept the protocols in this proposal.
• Contact the system programmer.

System programmer response
If the proposal that contains the mismatch is the one that should be accepted, notify the administrator of the remote security endpoint that you received a protocol mismatch and ask the administrator to ensure that they alter the remote configuration to propose the correct protocols.

Module
policy.cpp

Procedure name
None.

EZD0948I  IKE DETECTED MODIFY COMMAND ERROR : error

Explanation
While processing a MODIFY command, the IKE daemon received an error.
error is the error that was detected. Possible values are:

NO VERB
Indicates a required verb parameter was not specified on the MODIFY command.

TOO LONG
Indicates that the MODIFY command syntax was too long.

ZERO LENGTH
Indicates that the MODIFY command was missing required parameters.

INVALID COMMAND
Indicates that a verb that was not valid was used on the MODIFY command.

System action
The command is ignored; IKE daemon processing continues.

Operator response
Check the command syntax, and try the request again. If the command was too long, try using abbreviations where possible.

System programmer response
None.

Module
mdfysrvr.cpp

Procedure name
None.

EZD0950I  IKE CANNOT RETRIEVE MODIFY COMMAND DATA

Explanation
The Internet Key Exchange (IKE) daemon encountered an error while attempting to retrieve MODIFY command data from the communications area.
System action
The IKE daemon was unable to process the MODIFY command.

Operator response
Try the request again. If the failure persists, look for other OS messages about the failure.

System programmer response
None.

Module
mdfysrvr.c

Procedure name
None.

EZD0951I  No proposal data to lay out

Explanation
An IKE negotiation failed because the IKE daemon found no valid proposals.

System action
The Security Association (SA) negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received an empty proposal and ask the administrator to ensure that their policy is configured correctly.

Module
layout.cpp

Procedure name
None.

EZD0952I  Error on ioctl call (ioctl) : errno | errnojr | errortext

Explanation
The system was unable to successfully perform the specified ioctl call.

ioctl is the ioctl call that was called.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

errortext provides further information about why the ioctl call that was called failed or it describes the meaning of errno.
**System action**
The operation being performed failed; IKE daemon processing continues.

**Operator response**
Verify that the system stacks are started and operating correctly and reissue the command. If the error persists, contact the system programmer.

**System programmer response**
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

**Module**
isakmp_base_sa.cpp, simple_net.cpp, stackobj.cpp

**Procedure name**
None.

---

**EZD0953I**  Filter installation failed due to conflict with existing filter

**Explanation**
An IKE negotiation failed because the IKE daemon was unable to install a dynamic filter in the TCP/IP stack due to a conflict with an existing filter. This error occurs whenever an existing dynamic filter overlaps the new filter partially, or whenever an existing filter exactly matches the new filter. Because of the conflict, the stack cannot determine which filter should take precedence, so the new filter is rejected.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Change the properties of the tunnel so that the dynamic filter does not conflict with an existing dynamic filter, and restart the negotiation. If the error persists, contact the system programmer.

**System programmer response**
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

**Module**
stackobj.cpp

**Procedure name**
None.

---

**EZD0954I**  Transform transform_id : no encryption algorithm specified

**Explanation**
An IKE negotiation failed because the specified transform does not specify an encryption algorithm.
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

transform_id is the value used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to ensure that an encryption algorithm is specified for each transform.

Module
gen.cpp, oakley_lf.cpp

Procedure name
None.

EZD0956I  Transform transform_id : no Diffie-Hellman group identifier specified

Explanation
An IKE negotiation failed because the specified transform did not specify a Diffie-Hellman (DH) group identifier. Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

transform_id is the value used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to verify that they have a DH group specified for each transform.

Module
oakley_lf.cpp
EZD0957I  Phase 1 Security Association p1_said created successfully but an error at the coupling facility makes it unavailable for takeover

Explanation
The indicated Security Association (SA) was created and is fully functional. However, when the phase 1 information was passed into the coupling facility, an error occurred. This phase 1 SA is not available for takeover. p1_said is the ID of the specified SA.

System action
The SA is not available for takeover; IKE daemon processing continues.

Operator response
See the information about diagnosing sysplex-wide Security Association (SWSA) problems in z/OS Communications Server: IP Diagnosis Guide for information about correcting the problem at the coupling facility. After correcting the problem at the coupling facility, you can manually refresh the SA using the ipsec command to make it available for takeover. See the information about the managing network security in z/OS Communications Server: IP System Administrator’s Commands or issue the man ipsec command in a z/OS UNIX shell to obtain information about the ipsec command syntax and options. Alternatively, you can wait for the SA to be refreshed automatically according to the configured lifetime or lifesize settings.

System programmer response
None.

Module
phase1.cpp

EZD0958I  Unsupported protocol (protocol_id) for phase phase Security Association negotiation

Explanation
An IKE negotiation failed because the server encountered a protocol that is not valid in the current Security Association (SA) negotiation phase. Phase 1 negotiation supports only the IKE protocol and phase 2 supports only the ESP and AH protocols.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

protocol_id is the numerical value that identifies the unsupported protocol.

phase is the phase (1 or 2) that the negotiation was in when the error occurred.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.
**System programmer response**

Notify the administrator of the remote security endpoint about this error and ask the administrator to verify that they are using only the supported protocols for SA negotiations that are listed in **Explanation** above.

**Module**

doi.cpp, policy.cpp

**Procedure name**

None.

---

**EZD0959I**  Proposal ( proposal ) using protocol ( protocol_id ) is out of order - last proposal was last_proposal

**Explanation**

An IKE negotiation failed because the current proposal number is out of order. Proposals must be processed in monotonically increasing order.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*proposal* is the proposal number.

*protocol_id* is the protocol ID.

*last_proposal* is the number of the last proposal.

**System action**

The SA negotiation failed; IKE daemon processing continues.

**Operator response**

Contact the system programmer.

**System programmer response**

Notify the administrator of the remote security endpoint that you received a proposal out of order and ask the administrator to ensure that their proposal configuration is correct.

**Module**

policy.cpp

**Procedure name**

None.

---

**EZD0960I**  Phase 1 proposal contains non-IKE protocol

**Explanation**

An IKE negotiation failed because the current phase 1 proposal contains a non-IKE protocol. During phase 1, the only protocol allowed is IKE.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.
System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received a phase 1 message with a non-IKE protocol specified and to verify that their proposal configuration is correct.

Module
policy.cpp

Procedure name
None.

EZD0961I Proposal contains number extra bytes

Explanation
An IKE negotiation failed because the current proposal contains extra data. Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

number is the number of extra bytes that the proposal contains.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received a proposal that was not valid and ask the administrator to ensure that their policy is configured correctly.

Module
policy.cpp

Procedure name
None.

EZD0962I Responder mode in phase 1 policy does not match initiator mode (mode)

Explanation
The IKE negotiation failed because the mode specified for the responder in the policy database is not compatible with the role specified by the initiator of the Security Association (SA) negotiation.
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

mode is the mode that this negotiation requires. mode is either main or aggressive.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
When configured without the IBM Configuration Assistant for z/OS Communications Server, check the HowToRespond parameter in the KeyExchangeAction statement corresponding to this phase I negotiation. If the mode reported in mode should be allowed, change the HowToRespond parameter to allow mode. Otherwise, notify the administrator of the remote security endpoint that your policy does not allow this mode of negotiation. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

When configured with the IBM Configuration Assistant for z/OS Communications Server, edit the corresponding Connectivity Rule in the TCP/IP stack and check the Responder Mode setting on the Advanced IPSec: Dynamic Tunnels: Key Exchange Settings panel. If the mode reported in mode should be allowed, change the Responder Mode setting to allow mode. Otherwise, notify the administrator of the remote security endpoint that your policy does not allow this mode of negotiation. See the online helps in the GUI for additional information.

**System programmer response**
None.

**Module**
policy.cpp

**Procedure name**
None.

<table>
<thead>
<tr>
<th>EZD0963I</th>
<th>INTERNAL ERROR errid - UNABLE TO OBTAIN MEMORY OF SIZE size</th>
</tr>
</thead>
</table>

**Explanation**
The Internet Key Exchange (IKE) daemon was unable to obtain the required amount of memory storage.

errid is a unique ID for this error.

size is the amount of storage the server attempted to obtain.

**System action**
Results are unpredictable.

**Operator response**
Ensure that there is enough memory available on the system and try the operation again.

**System programmer response**
None.

**Module**
Numerous
EZD0964I  IKE CANNOT KILL MAIN THREAD : errno | errnojr | description

Explanation
The Internet Key Exchange (IKE) daemon received a STOP command but was unable to kill the main thread. The most likely reason is that the MAIN thread no longer exists.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

description describes the meaning of errno.

System action
The command failed.

Operator response
None.

System programmer response
None.

Module
mdfysrvr.c

EZD0965I  Validity check or authentication failure occurred using a shared key

Explanation
A validity check failure or authentication failure occurred on a message for a phase 1 Security Association configured for shared key authentication. This indicates a likely mismatch between the locally configured shared key value and the shared key value configured on the remote security endpoint.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA) including the applicable KeyExchangeRule statement. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Confirm that the configured shared Key value matches the value configured on the remote security endpoint.

When configured without the IBM Configuration Assistant for z/OS Communications Server, confirm that the SharedKey value on the applicable KeyExchangeRule statement matches the key value that is configured on the remote security endpoint. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Communications Server: IP Configuration Reference for more information about the KeyExchangeRule statement and the SharedKey parameter.

When configured with the IBM Configuration Assistant for z/OS Communications Server, confirm that the Shared Key value on the applicable Connectivity Rule matches the key value that is configured on the remote security endpoint. See the online helps in the GUI for additional information.

**System programmer response**
None.

**Module**
oakley_kep.cpp

**Procedure name**
None.

**EZD0966I IKE CANNOT INITIALIZE MODIFY COMMAND QUEUE**

**Explanation**
The Internet Key Exchange (IKE) daemon was unable to initialize the MODIFY command queue.

**System action**
The MODIFY command service is not available; IKE daemon processing continues.

**Operator response**
If you require the MODIFY command service, restart the IKE daemon. If the problem persists, contact the system programmer.

**System programmer response**
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

**Module**
mdfysrvr.c

**Procedure name**
None.

**EZD0967I IKE RELEASE ver.rel.mod SERVICE LEVEL level CREATED ON date**

**Explanation**
This message is issued when the IKE daemon is started.

*ver.rel.mod* is the z/OS version, release, and modification level.

*level* is the service level of the IKE daemon.

*date* is the date this version of the IKE daemon was created.

**System action**
IKE daemon processing continues.
Operator response
None.

System programmer response
None.

Module
fw_initterm.c

Procedure name
None.

---

**EZD0968I**  Unknown or incorrect address type (address_type) in Security Association address

**Explanation**
The specified address type in the SA is unsupported.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:

- **address_type**
  The number of the unsupported address type

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Request that the administrator of the remote security endpoint check their policy and ensure that only supported addresses are being used.

---

**Module**
sa_addr.cpp

**Procedure name**
None.

**EZD0969I**  Proposal too short or security parameter index (SPI) size too large (reason)

**Explanation**
An IKE negotiation failed because the proposal does not contain enough data or the SPI size in the transform is too large.

Additional diagnostic messages with the same message instance number will be issued identifying the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*reason* indicates which parameter was incorrect. *reason* is either *spi* or *proposal*.
**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Notify the administrator of the remote security endpoint that you received a payload that was not valid and ask the administrator to ensure that their policy is configured correctly.

**Module**
policy.cpp

**Procedure name**
None.

**EZD0970I C/C++ runtime library function call function failure : errid - errno | errnojr | description**

**Explanation**
A C/C++ run-time library function call returned an error. This error should not occur during normal processing. function is the function call that failed. See the z/OS XL C/C++ Runtime Library Reference for more information. errid is a unique ID for this error. errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errno) information in z/OS UNIX System Services Messages and Codes. errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errno) information in z/OS UNIX System Services Messages and Codes, where the reason codes are listed. description describes the meaning of errno.

**System action**
The operation being performed failed; IKE daemon processing continues.

**Operator response**
Correct the error indicated by errno, errnojr, and description and restart the IKE daemon if necessary.

**System programmer response**
None.

**Module**
cp_convert.c, fw_initterm.c, fwmsgq.c, main.cpp, sa.cpp, simple_net.cpp, simple_timer.cpp, simple_ureq.cpp, stackmgr.cpp, stckobj.cpp, syslogc.c, verifyq.cpp

**Procedure name**
None.

**EZD0971I Trying to use phase 2 Security Association but phase 1 key exchange is not complete**
**Explanation**

The Internet Key Exchange (IKE) daemon attempted to send a notification with a phase 2 security association (SA), but the phase 1 key exchange is not complete.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

**System action**

The message cannot be sent until the phase 1 key exchange is complete; the IKE daemon continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

doi.cpp

**Procedure name**

None.

**EZD0973I**  
Responder-Lifetime notification payload received for phase security association and is ignored

**Explanation**

The Internet Key Exchange (IKE) daemon received a Responder-Lifetime notification message. The Internet Key Exchange (IKE) daemon ignores Responder-Lifetime notification messages and maintains the original lifetime value.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*phase* is either 1 for an IKE Security Association or 2 for a dynamic Security Association.

**System action**

IKE daemon processing continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

doi.cpp, oakley_phaseII.cpp

**Procedure name**

None.
**EZD0974I**  
Attribute length *(attribute_length)* is not valid - expected *(valid_length)*

**Explanation**

The Internet Key Exchange (IKE) daemon received a notification message with an incorrect attribute length. Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*attribute_length* is the length that was received.

*valid_length* is the length that was expected.

**System action**

The message request failed; IKE daemon processing continues.

**Operator response**

Notification messages are not critical to IKE operation, but if this message is seen repeatedly, notify the administrator of the remote security endpoint that you are receiving messages that are not valid.

**System programmer response**

None.

**Module**

doi.cpp

**Procedure name**

None.

**EZD0975I**  
Unknown message type *(message_type)*

**Explanation**

The Internet Key Exchange (IKE) daemon received a message type that is not supported. Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*message_type* is the number of the message type received.

**System action**

The message cannot be processed; IKE daemon processing continues.

**Operator response**

None. However if this message is seen often, contact the system programmer.

**System programmer response**

None. However if this message is seen often, notify the administrator of the remote security endpoint that you are receiving messages that are not valid.

**Module**

doi.cpp
**EZD0977I**  IKE CANNOT START BECAUSE IT IS ALREADY RUNNING

**Explanation**
There can be only one active Internet Key Exchange (IKE) daemon. A second IKE daemon cannot be started when one is already running.

**System action**
The IKE daemon that attempted to start ends; the running IKE daemon continues.

**Operator response**
Use the STOP command to stop the active IKE daemon and try the request again.

**System programmer response**
None.

**Module**
fw_initterm.c

---

**EZD0978I**  Initiator must specify exchange type

**Explanation**
An IKE negotiation failed because the initiator of the negotiation did not specify an exchange type. The initiator must specify whether to use main or aggressive mode.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Notify the remote security endpoint that you received a request to start a negotiation, but the exchange type was not specified.

**Module**
policy.cpp

---
EZD0979I  Refresh of phase 1 Security Association p1_SA_ID could not be performed

Explanation
The Internet Key Exchange (IKE) daemon could not refresh the specified phase 1 Security Association (SA) because the IKE message queue could not be accessed.

Additional diagnostic messages that have the same message instance value will be issued to further identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

p1_SA_ID is the phase 1 SA ID of this SA.

System action
The SA is not refreshed; IKE daemon processing continues.

Operator response
No action is required, but the SA will expire after the normal lifetime or life-size has been exceeded. To refresh the SA manually, use the `ipsec -k refresh` command.

See the information about the managing network security information in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

System programmer response
None.

Module
isakmp_base_sa.cpp

Procedure name
None.

EZD0980I  No transform specified in Key Exchange Protocol initialization

Explanation
The Internet Key Exchange (IKE) daemon received a message to start a Key Exchange Protocol (KEP) but no transform was specified.

System action
The request failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to verify that their policy is configured correctly.

Module
genkep.cpp
Procedure name
None.

EZD0981I   Unable to set ID - ID mismatch

Explanation
The Internet Key Exchange (IKE) daemon could not set the ID of a security endpoint because the data did not match what was expected. This might occur when initiating a phase 1 security association (SA) negotiation and the responder used a different identity than was expected based on local configuration.

Additional diagnostic messages that have the same message instance number will be issued to further identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The request failed; IKE daemon processing continues.

Operator response
Verify that the LocalSecurityEndpoint and RemoteSecurityEndpoint statements have valid identity strings and are defined correctly. Review the configuration of the IKE peer to confirm that it uses the same identity as is locally configured.

System programmer response
None.

Module
do.i.cpp

Procedure name
None.

EZD0982I   Payload length error

Explanation
The Internet Key Exchange (IKE) daemon encountered a payload that was not the correct size.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The request failed and the SA negotiation might fail; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received a payload that was not valid and ask the administrator to ensure that their policy is configured correctly.
Module
isakmp_base_sa.cpp, oakley_kep.cpp, pksig.cpp, simple_ureq.cpp

Procedure name
None.

EZD0983I  Unknown request type (request_id)

Explanation
The Internet Key Exchange (IKE) daemon received a request to perform an unknown action.
request_id is the number that identifies the request.

System action
The request failed; IKE daemon processing continues.

Operator response
Ensure that the request entered is valid and try the operation again.

System programmer response
None.

Module
anchor_ureq.cpp, simple_ureq.cpp

Procedure name
None.

EZD0984I  IKE function locid - function failed : val1 | val2 | val3

Explanation
An IKE function did not complete successfully because of an error condition in the function specified.
Additional diagnostic messages that have the same message instance number will be issued to further identify
the error. The message instance number precedes the message number in the log output and is used to group
related messages from the IKE daemon.
locid identifies the location of this error.
function is the IKE function that did not complete successfully.
val1 is internal error information or 0 if not available.
val2 is internal error information or 0 if not available.
val3 is internal error information or blank if not available.

System action
The specified function failed; IKE daemon processing continues.

Operator response
Look for other messages that have the same message instance number and follow the instructions documented
in those messages. Because these messages are usually the result of other errors that are being encountered,
they can be turned off by not setting the VERBOSE bit in the IkeSyslogLevel parameter on the IkeConfig
statement. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IkeConfig statement and IKE daemon configuration file.

System programmer response
None.

Module
various

Procedure name
None.

EZD0985I  No proposal chosen

Explanation
An IKE negotiation failed because no proposal in the offer was accepted by the local security endpoint. Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Change the local policy configuration or ask the remote security end point administrator to change their policy so that a proposal can be accepted.

When configured without the IBM Configuration Assistant for z/OS Communications Server, see the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

When configured with the IBM Configuration Assistant for z/OS Communications Server, see the online helps in the GUI for additional information.

System programmer response
None.

Module
policy.cpp

Procedure name
None.

EZD0986I  IKE IS NOT APF AUTHORIZED

Explanation
An attempt was made to start the Internet Key Exchange (IKE) daemon application, but the application is not APF authorized. APF authorization is required to execute the IKE daemon.

System action
The IKE daemon ends.
Operator response
Contact the system programmer.

System programmer response
Ensure that the IKE daemon resides in an APF-authorized library.

Module
fw_initterm.c

Procedure name
None.

EZD0987I Certificate payload or request received but certificates are not supported

Explanation
The Internet Key Exchange (IKE) daemon encountered a certificate payload or a certificate request in a message, but the server is not configured to support certificates.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The certificate cannot be used and the negotiation will probably fail; the IKE daemon continues.

Operator response
If you want RSA signature authentication, verify that the Key ring name setting is correct. Otherwise, ensure that the remote security endpoint administrator uses the shared key method of authentication.

When configured without the IBM Configuration Assistant for z/OS Communications Server, the Key ring database setting is specified on the IkeConfig statement with the KeyRing parameter. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IkeConfig statement.

When configured with the IBM Configuration Assistant for z/OS Communications Server, the Key ring database name is configured on the IPSec: IKE Daemon Settings panel. See the online helps in the GUI for additional information.

System programmer response
None.

Module
oakley_kep.cpp

Procedure name
None.

EZD0988I Rcookie is zero - message ID is non-zero
Explanation
During an IKE message exchange, the IKE daemon received a message that had an Rcookie with a value of 0, but the message ID was nonzero. A nonzero message ID indicates a phase 2 negotiation, in which case the Rcookie is required to be nonzero.

System action
The message was not processed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about the error and ask the administrator to ensure that their policy is configured correctly.

Module
anchor_msg.cpp

Procedure name
None.

EZD0989I Transform transform_id : Diffie-Hellman group (DH_group) is not supported

Explanation
The Internet Key Exchange (IKE) daemon detected a Diffie-Hellman (DH) group that is not supported. Only groups 1, 2, 5, and 14 are supported.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

transform_id is the value used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

DH_group is the ID of the unsupported DH group. DH groups are specified on the DHGroup parameter on a KeyExchangeOffer Statement.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about the error and ensure that their DH group identifiers are specified correctly.

Module
gen.cpp
The Internet Key Exchange (IKE) daemon successfully accessed the SAF key ring to support the RSA signature mode of authentication. This key ring was specified by the KeyRing parameter of the IkeConfig statement.

In the message text:

**stackname**

The name of the stack for which RSA signature mode of authentication is available.

The IKE daemon will support RSA signature mode of authentication for the stack indicated by the **stackname** value; the IKE daemon continues.

**Example**

```
EZD0990I The IKE daemon is set up to support RSA signature mode of authentication for stack TCPCS using the local keyring
```

```
EZD0991I Transform transform_id : unsupported authentication type (auth_type) for phase 1 Security Association
```
Explanation
An IKE phase 1 Security Association (SA) negotiation failed because the authentication type is not supported.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

transform_id is the value used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement.

auth_type is the number of the unsupported authentication type. The valid authentication types are PreshareKey and RsaSignature, which are specified on the HowToAuthPeers parameter of an KeyExchangeOffer statement.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to ensure that only pre-shared keys or RSA signature is being used for peer authentication.

Module
gen.cpp

Procedure name
None.

EZD0992I IKE daemon running swappable : errno errno ( description ) errnojr errnojr

Explanation
An error occurred when the IKE daemon attempted to set itself as non-swappable.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

description describes the meaning of errno.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
IKE daemon processing continues as a swappable process.

Operator response
None.

System programmer response
Running the process as swappable might affect the IKE daemon performance. Correct the problem, if possible, and restart the IKE daemon.
Module
main.cpp

Procedure name
None.

**EZD0993I** Transform *transform_id* : Unable to layout *protocol* transform header

**Explanation**
An IKE negotiation failed because the server cannot build the transform header for the specified protocol. This error is due to either the transform header being empty or a memory allocation failure.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*transform_id* is the value used to identify this transform in an IKE proposal. Supported transforms for IKE SAs are described in Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Phase 1 transforms are specified on a KeyExchangeOffer statement, and phase 2 transforms are specified on an IpDataOffer statement. Phase 2 transforms specify the encapsulation mode on the HowToExcap statement.

*protocol* is a string representing the protocol. *protocol* is either **AH**, **ESP**, or **OAKLEY**.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
If the error was due to a memory allocation failure, fix the error and try again. Otherwise, contact the system programmer.

**System programmer response**
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

**Module**
ah lf.cpp, esp lf.cpp, oakley lf.cpp

**Procedure name**
None.

**EZD0994I** Unable to initialize anchor

**Explanation**
During initialization, the IKE daemon was unable to initialize the anchor that is the main IKE control object.

Additional diagnostic messages that have the same message instance number will be issued to identify the error. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

**System action**
The IKE daemon ends.
Operator response
See the User or Operator response in the other messages that have the same message instance number to correct the error.

System programmer response
None.

Module
main.cpp

Procedure name
None.

**EZD0995I**  Phase 1 Security Association sa_id is expired

Explanation
The Internet Key Exchange (IKE) daemon detected a request to access a phase 1 Security Association (SA) that has expired.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

sa_id is the phase 1 SA ID.

System action
The request failed; IKE daemon processing continues.

Operator response
To satisfy the request, the SA must be refreshed or re-started. Use the `ipsec -k refresh` or `ipsec -k activate` command to refresh or activate a phase 1 SA. See the information about the managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

System programmer response
None.

Module
oakley_kep.cpp, sa.cpp

Procedure name
None.

**EZD0996I**  Unsupported payload type (type)

Explanation
The Internet Key Exchange (IKE) daemon detected a payload of an unsupported type.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.
type is the numerical representation of the unsupported payload type.

System action
The request failed because the message was not processed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received an unsupported payload type and ask the administrator to ensure that their policy is configured correctly.

Module
msg.cpp, oakley_kep.cpp

Procedure name
None.

EZD0998I Unable to authenticate IKE message

Explanation
An IKE request or negotiation failed because the hash verification of the message failed.
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The IKE request or SA negotiation failed; IKE daemon processing continues.

Operator response
Try the operation again. If errors continue, contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you are receiving messages that you are unable to authenticate and ask the administrator to ensure that they are using a valid hash algorithm.

Module
oakley_kep.cpp

Procedure name
None.
Chapter 12. EZD1xxx messages

EZD1005I  Too many life attributes specified

Explanation
The Internet Key Exchange (IKE) daemon received a request to modify the life of a security association (SA) and detected an attribute list that contains too many attributes. Valid attributes are LIFE-TYPE and LIFE SIZE.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The request failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received this message and ask the administrator to verify that their policy is configured correctly.

Module
oakley_kep.cpp

Procedure name
None.

EZD1006I  No relevant payload was received

Explanation
The Internet Key Exchange (IKE) daemon detected a message that does not contain a payload required to handle the current request.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted security association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The request failed and the SA negotiation might fail; IKE daemon processing continues.

Operator response
If the SA negotiation fails, notify the administrator of the remote security endpoint that you received a payload containing no relevant information.

System programmer response
None.
Module
oakley_kep.cpp, phase1.cpp

Procedure name
None.

EZD1007I Received message is not encrypted

Explanation
The Internet Key Exchange (IKE) daemon detected a message that was not encrypted but should have been. Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The request failed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received a message that should have been encrypted but was not and ask the administrator to ensure that their policy is configured correctly.

Module
oakley_kep.cpp

Procedure name
None.

EZD1008I errid System SSL CMS call function failure : gsk_rc description

Explanation
A call to the System SSL Certificate Management Services (CMS) API returned an error. This error should not occur during normal processing.

errid is the unique ID for this error.

function is the API call that failed. The System SSL CMS API is documented in z/OS Cryptographic Services System SSL Programming.

gsk_rc is the hexadecimal CMS status code. See the information about the CMS status codes in z/OS Cryptographic Services System SSL Programming.

description describes the meaning of gsk_rc.

System action
The operation being performed fails; IKE daemon processing continues.

Operator response
Use gsk_rc and the description provided to fix the error.
System programmer response
None.

Module
asn_utils.cpp, cacache.cpp, cert_rep.cpp, certcache.cpp, pki390.cpp, key_agree.c

Procedure name
None.

EZD1009I  Received message is a replay message

Explanation
The Internet Key Exchange (IKE) daemon detected a message that was already processed.

Additional diagnostic messages that have the same message instance number will be issued to identify the
impacted Security Association (SA). The message instance number precedes the message number in the log
output and is used to group related messages from the IKE daemon.

System action
The message is not processed; IKE daemon processing continues.

Operator response
Replays are not errors and can be caused by lost packets, network congestion. Usually, no action is required. If
there are a large number of replays, notify the administrator of the remote security endpoint that you are
receiving a large number of replays.

System programmer response
None.

Module
oakley_kep.cpp, oakley_phaseII.cpp, phase1.cpp

Procedure name
None.

EZD1010I  IKE message exchange is in unknown state (state) for mode phase phase Security
Association

Explanation
The Internet Key Exchange (IKE) daemon detected a message that cannot be processed for a security
association (SA) in the current state.

Additional diagnostic messages that have the same message instance number will be issued to identify the
impacted SA. The message instance number precedes the message number in the log output and is used to
group related messages from the IKE daemon.

state is the current state of the SA negotiation. state is either unknown or wait for key exchange.
mode is the mode in which the daemon is acting. mode is either initiator or responder.
phase is the phase of this SA. phase is either 1 or 2.
**System action**
The message cannot be processed; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Notify the administrator of the remote security endpoint that you are receiving messages that are not valid for the state and mode that is current for this SA negotiation and ask the administrator to ensure that their policy is configured correctly.

**Module**
oakley_phaseII.cpp, phase1.cpp

**Procedure name**
None.

**EZD1011I Duplicate phase 2 payload payload encountered**

**Explanation**
The Internet Key Exchange (IKE) daemon detected a payload that was already received or received more payloads of this type than expected.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

payload is the type of duplicate payload. payload can be one of the following:

- Key Exchange
- ID
- NONCE

**System action**
The message was not processed; IKE daemon processing continues.

**Operator response**
If a large number of these messages are issued, notify the administrator of the remote security endpoint that you are receiving duplicate messages.

**System programmer response**
None.

**Module**
oakley_phaseII.cpp

**Procedure name**
None.

**EZD1012I Unexpected payload type (payload)**
Explanation
The Internet Key Exchange (IKE) daemon detected a payload type that is not supported or not expected in the current stage of processing.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

payload is the type of payload found. payload is one of the following:

- Key Exchange
- ID
- NONCE
- other

System action
The message was not processed; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you received an unexpected payload type and ask the administrator to ensure that their policy is configured correctly.

Module
oakley_phaseII.cpp

Procedure name
None.

EZD1013I  Unexpected phase 2 message - not a replay

Explanation
The Internet Key Exchange (IKE) daemon received a message during phase 2 processing that was not expected and was not a replay message.

System action
The message is ignored; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that you are receiving unexpected phase 2 messages during this Security Association negotiation and ask the administrator to verify that their policy is configured correctly.

Module
oakley_phaseII.cpp
**EZD1014I  No payload payload received**

**Explanation**
During an IKE message exchange, the IKE daemon received a message indicating that a payload of the specified type should have already been processed, but the payload had not been received in any previous message.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*payload* is the type of payload received. *payload* can be one of the following:

- **NONCE**
- **ID**
- **Key Exchange**
- **Common**

**System action**
The message is not processed; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Notify the administrator of the remote security endpoint that you did not receive the payload mentioned above and ask the administrator to verify that their policy is configured correctly.

**Module**
oakley_phaseII.cpp, phase1.cpp, sa.cpp

**EZD1015I  Unknown object (value)**

**Explanation**
The Internet Key Exchange (IKE) daemon encountered an object that is not supported.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*object* is the type of object being operated on. *object* can be one of the following:

- **DOI**
- **notify type**
- **transform**
- **authAlg**

*value* is the numerical representation of the object value.
**System action**

If object is **DOI** or **notify type**, the notification was not processed. Otherwise, the SA negotiation failed. IKE daemon processing continues.

**Operator response**

Contact the system programmer.

**System programmer response**

Notify the administrator of the remote security endpoint that you are receiving data that is not valid in IKE messages and ask the administrator to ensure that their policy is configured correctly.

**Module**

infoXchg.cpp, oakley_phaseII.cpp

**Procedure name**

None.

---

**EZD1018I** Could not resolve hostname for InitiateToLocation (hostname)

**Explanation**

The Internet Key Exchange (IKE) daemon failed to resolve the specified host name. 

hostname is the host name that could not be resolved.

**System action**

The request failed; IKE daemon processing continues.

**Operator response**

Verify that the host name is correct and that your domain name server is working correctly.

When configured without the IBM Configuration Assistant for z/OS Communications Server, the host name setting is configured on the InitiateToLocation parameter of a LocalStartAction statement. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about the InitiateToLocation parameter.

When configured with the IBM Configuration Assistant for z/OS Communications Server, to verify that the host name setting is correct, edit the corresponding Connectivity Rule in the GUI. For Host to Gateway or Gateway to Gateway topologies, this setting is found on the Remote Security End Point panel. For Host to Host or Gateway to Host, this setting is found on the Advanced IPSec: Dynamic Tunnels: Local Activation panel. See the online helps in the GUI for additional information.

**System programmer response**

None.

**Module**

policymgr.cpp

**Procedure name**

None.

---

**EZD1019I** Could not open certificate repository (name) (description) (gsk_rc)
Explanation
The Internet Key Exchange (IKE) daemon was unable to open the certificate repository identified by the key ring database setting.

name is the name of the certificate repository (key ring database) that IKE was unable to open.

description describes the meaning of gsk_rc.

gsk_rc is the hexadecimal Certificate Management Services (CMS) status code. See the information about the CMS status codes in z/OS Cryptographic Services System SSL Programming.

System action
The IKE daemon will not support RSA signature mode authentication; IKE daemon processing continues.

Operator response
Ensure that the repository name is defined correctly and that the user under which the IKE daemon was started is authorized to access the repository.

When configured without the IBM Configuration Assistant for z/OS Communications Server, the certificate repository name is set on the KeyRing parameter of the IkeConfig statement. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information.

When configured with the IBM Configuration Assistant for z/OS Communications Server, the certificate repository name is set in the key ring database name located in the IPSec: IKE Daemon Settings panel.

System programmer response
None.

Module
cert_rep.cpp

Procedure name
None.

EZD1020I Incorrect payload size (payload_length / length-expected)

Explanation
The Internet Key Exchange (IKE) daemon detected a payload that is not the correct size for the expected payload type.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

payload_length is the length of the payload received.

length-expected is the length the payload should have been.

System action
The request failed; IKE daemon processing continues.

Operator response
Contact the system programmer.
System programmer response
Notify the administrator of the remote security endpoint that you received a payload with an incorrect size and ask the administrator to verify that their policy is configured correctly.

Module
msg.cpp, oakley_phaseII.cpp

Procedure name
None.

EZD1021I   No proposal chosen with KeyExchangeRule ( rule ) and KeyExchangeAction ( action )

Explanation
An IKE phase 1 negotiation failed because no proposal in the offer was accepted by the local security endpoint. Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

- rule is the rule name.
  - In the policy agent configuration file, rule is the name of a KeyExchangeRule statement in the policy agent configuration.
  - When configured with the IBM Configuration Assistant for z/OS Communications Server, rule corresponds to the name of a Connectivity Rule. rule also contains a suffix appended to the Connectivity Rule name to guarantee uniqueness.

- action is the action name.
  - In the policy agent configuration file, action is the name of a KeyExchangeAction statement in the policy agent configuration file.
  - When configured with the IBM Configuration Assistant for z/OS Communications Server, action corresponds to the name of a Connectivity Rule. The rule name also contains a numeric suffix appended to the Connectivity Rule name to guarantee uniqueness.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
This message will be accompanied by one or more EZD1093I messages. See the User or Operator Response in EZD1093I for information about resolving this negotiation failure.

System programmer response
None.

Module
policy.cpp

Procedure name
None.

EZD1022I   No proposal chosen with IpFilterRule ( rule ) and IpDynVpnAction ( action )
Explanation
An IKE phase 2 negotiation failed because no proposal in the offer was accepted by the local security endpoint.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

rule is the name of a filter rule.

• In the policy agent configuration file, rule is the name of an IpFilterRule statement.
• When configured with the IBM Configuration Assistant for z/OS Communications Server, rule is the name of a Connectivity Rule in GUI. rule also contains a numeric suffix appended to the Connectivity Rule name to guarantee uniqueness.

action is the action name.

• In the policy agent configuration file, action is the name specified on the applicable IpDynVpnAction statement.
• When configured with the IBM Configuration Assistant for z/OS Communications Server, the action name corresponds to the name of the security level in the GUI. The action name also contains a numeric suffix appended to the security level name to guarantee uniqueness.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
This message will be accompanied by one or more EZD1093I messages. See the User or Operator Response in EZD1093I for information about resolving this negotiation failure.

System programmer response
None.

Module
policy.cpp

Procedure name
None.

EZD1025I Cannot be an initiator of a phase 2 Security Association negotiation

Explanation
The local IKE daemon is attempting to initiate a phase 2 security association (SA), and the local policy specifies that it can only act as a responder.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Check the server's configuration for phase 2 activation.
When configured without the IBM Configuration Assistant for z/OS Communications Server, the IKE daemon's phase 2 initiation role is set on the Initiation parameter in the IpDynVpnAction statement for this SA. If the local IKE server should be able to initiate the negotiation for this SA, then change the server's Initiation role in the appropriate IpDynVpnAction statement to **LocalOnly** or **Either**. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

When configured with the IBM Configuration Assistant for z/OS Communications Server, edit the corresponding Connectivity Rule in the GUI and check the Advanced IPSec: Dynamic Tunnel: How to Activate panel to see if local activation of phase 2 tunnels is allowed. See the online helps in the GUI for additional information.

**System programmer response**

None.

**Module**

policy.cpp

**Procedure name**

None.

**EZD1026I Cannot be a responder in a phase 2 Security Association negotiation**

**Explanation**

The local IKE daemon is attempting to respond to a phase 2 security association (SA) negotiation request and the local policy specifies that it can act only as an initiator.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

**System action**

The SA negotiation failed; IKE daemon processing continues.

**Operator response**

Check the server's configuration for phase 2 activation.

When configured without the IBM Configuration Assistant for z/OS Communications Server, the IKE daemon's phase 2 initiation role is set on the Initiation parameter in the IpDynVpnAction statement for this SA. If the local IKE server should be able to be a responder in the negotiation for this SA, then change the server's Initiation role in the appropriate IpDynVpnAction statement to **RemoteOnly** or **Either**. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

When configured with the IBM Configuration Assistant for z/OS Communications Server, edit the corresponding Connectivity Rule in the GUI and check the Advanced IPSec: Dynamic Tunnel: How to Activate panel to see if remote activation of phase 2 tunnels is allowed. See the online helps in the GUI for additional information.

**System programmer response**

None.

**Module**

policy.cpp
**EZD1027I Unsupported security endpoint identity type id_type**

**Explanation**
An IKE request or negotiation failed because a security endpoint identity of an unsupported type was encountered. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for descriptions of LocalSecurityEndpoint and RemoteSecurityEndpoint statements.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

`id_type` is the security endpoint identity type.

**System action**
The IKE request or SA negotiation failed; IKE daemon processing continues.

**Operator response**
Ensure that the initiator's and the responder's IDs are defined correctly and try the request again.

**System programmer response**
None.

**Module**
pki390.cpp, fwconvert.c

---

**EZD1029I Certificate usage value (usage_value) not supported**

**Explanation**
An IKE negotiation failed because the certificate usage value is not supported. The values currently supported are the integer representation of the digital signature flag or `any`. This error occurred when verifying the signature of the remote IKE daemon certificate.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

`usage_value` is the number that represents the certificate usage value.

**System action**
The certificate could not be verified and the message was not processed; IKE daemon processing continues.

**Operator response**
Ensure that the certificates of the responder of the IKE exchange are configured correctly. This error might be due to the remote security endpoint sending an unsupported certificate; ensure the remote security endpoint is using valid certificates.
The IKE daemon is not set up to support RSA signature mode of authentication for stack stackname using the local keyring

Explanation
The Internet Key Exchange (IKE) daemon encountered an error while processing certificates located on the specified SAF key ring.

In the message text:

stackname
The name of the stack for which RSA signature mode of authentication is not available.

System action
The IKE daemon will not support RSA signature mode of authentication for the stack indicated by the stackname value; the IKE daemon continues.

Operator response
Contact the system programmer.

System programmer response
If RSA signature authentication is to be used, verify that the key ring database name is correct.

When configured without the IBM Configuration Assistant for z/OS Communications Server, the key ring database setting is specified on the IkeConfig statement with the KeyRing parameter. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IkeConfig statement and the IKE daemon configuration file.

When configured with the IBM Configuration Assistant for z/OS Communications Server, the key ring database name is configured on the IPSec: IKE Daemon Settings panel. See the online helps in the GUI for additional information.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
cert_mgr.cpp
Routing code
10

Descriptor code
12

Example

EZD1030I The IKE daemon is not set up to support RSA signature mode of authentication for stack TCPCS using the local keyring

EZD1031I DVIPA dvipa_addr added to the IKE daemon for stack stackname

Explanation
The Internet Key Exchange (IKE) daemon received an event from the TCP/IP stack to add a dynamic virtual IP address (DVIPA). The IKE daemon successfully added the DVIPA address to the stack indicated.

dvipa_addr is the dynamic virtual IP address that was added.
stackname is the stack to which the DVIPA was added.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

Module
anchor.cpp

Procedure name
None.

EZD1032I DVIPA dvipa_addr deleted from the IKE daemon for stack stackname

Explanation
The Internet Key Exchange (IKE) daemon received an event from the TCP/IP stack to delete a dynamic virtual IP address (DVIPA). The IKE daemon successfully deleted the DVIPA address from the stack indicated.

dvipa_addr is the dynamic virtual IP address that was deleted.
stackname is the stack from which the DVIPA was deleted.

System action
IKE daemon processing continues.

Operator response
None.
System programmer response
None.

Module
anchor.cpp

Procedure name
None.

**EZD1033I** IKE configuration file parameter *pname* contains value *val* which exceeds the maximum allowed character length *max_len* on line *linenum*

**Explanation**
The IKE configuration file contains a parameter that contains a value that exceeds the maximum character length allowed.

In the message text:
- **pname**: The parameter name.
- **val**: The parameter value that has a length that is too long.
- **max_len**: The maximum character length allowed for the parameter value.
- **linenum**: The line of the configuration file where the parameter was found.

**System action**
If the error occurs during IKE startup, IKE configuration file processing ends, and the IKE daemon ends. If the error occurs as a result of a MODIFY REFRESH command, IKE configuration file processing ends, but IKE remains active. IKE configuration retains all values configured prior to the MODIFY REFRESH command.

**Operator response**
Contact the system programmer.

**System programmer response**
Replace the value for the parameter specified by the *pname* value with one that does not exceed maximum length.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
ike_config.cpp
Routing code
10

Descriptor code
12

Example

EZD1033I IKE configuration file parameter SupportedCertAuth contains value A2345678901234567890123 which exceeds the maximum allowed character length 32 on line 6

EZD1034I Phase 1 Security Association for DVIPA dvipa_addr is already re-established with its remote security endpoint ip_addr

Explanation
The Internet Key Exchange (IKE) daemon already re-established the phase 1 Security Association (SA) with its remote security endpoint. This phase 1 SA already existed prior to the dynamic virtual IP address (DVIPA) takeover or giveback.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

dvipa_addr is the dynamic virtual IP address.

ip_addr is the IP address of the remote security endpoint.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

Module
anchor_ureq.cpp

Procedure name
None.

EZD1035I Certificate cannot be used for RSA signature mode of authentication

Explanation
IKE encountered a certificate that cannot be used for RSA signature mode of authentication; the IKE daemon currently supports only RSA signing IKEv1.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.
**System action**
The certificate cannot be used and the negotiation will fail if the certificate is an end-entity certificate; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Notify the administrator of the remote security endpoint about the error and ask the administrator to verify that certificate sent to the IKE daemon for IKEv1 are using RSA signature mode. The administrator of the remote security endpoint should also verify that the key usage and the extended key usage extensions of the certificates that were sent support the creation and verification of digital signatures in an IKE flow. When the key usage extension is present, either the digital signature bit or the nonrepudiation bit must be set. When the key usage extension is present it must allow either any usage or usage with IKE.

**Module**
pki390.cpp

**Procedure name**
None.

**EZD1036I** Phase 2 Security Association for DVIPA dvipa_address is not re-established with remote security endpoint ip_addr

**Explanation**
The Internet Key Exchange (IKE) daemon detected that the phase 2 Security Association (SA) for the dynamic virtual IP address (DVIPA) was not re-established with its remote security endpoint. An error might have occurred during the phase 2 SA negotiation process.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

dvipa_address is the dynamic virtual IP address.
ip_addr is the IP address of the remote security endpoint.

**System action**
The IKE phase 2 SA was not re-established for the DVIPA; IKE daemon processing continues.

**Operator response**
Check the IKE log for an entry indicating what the phase 2 SA negotiation error might be. Also, try to manually establish a phase 2 SA with the remote security endpoint by issuing the `ipsec -y activate` command.

See the information about the managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

**System programmer response**
None.

**Module**
phaseII_sa.cpp
EZD1037I The IKE daemon has no matching certificate entry for the specified LocalSecurityEndpoint identity (id_string) and certificate authority (X.500_string)

Explanation
The IKE message cannot be processed because no matching certificate entry was found. This error occurred while searching for a certificate that matched the LocalSecurityEndpoint ID and was signed by the CA that was requested by the remote security endpoint.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

id_string is a string representation of the LocalSecurityEndpoint ID.
X.500_string is the certificate authority.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Verify that the Local Security Endpoint Identity is correct. If it is correct, obtain a certificate with the expected ID of the local IKE server. When the certificate is obtained, add it to the IKE key ring with RACDCERT.

When configured without the IBM Configuration Assistant for z/OS Communications Server, the Local Security Endpoint is set on the LocalSecurityEndpoint statement. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about the LocalSecurityEndpoint statement.

When configured with the IBM Configuration Assistant for z/OS Communications Server, edit the corresponding Connectivity Rule in the GUI and verify that the Local Security Endpoint Identify is correct. See the online helps in the GUI for additional information.

System programmer response
None.

Module
pki390.cpp

EZD1038I Remote security endpoint's certificate is not valid because the security association's lifetime is not in the certificate's lifetime

Explanation
The Internet Key Exchange (IKE) daemon could not verify that the remote security endpoint certificate timeframe was valid. The current time is either before the certificate notBefore time value, or the lifetime of the Security Association (SA) is beyond the certificate notAfter time value.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.
System action
The certificate cannot be used, and the SA negotiation will probably fail; IKE daemon processing continues.

Operator response
Verify that the remote security endpoint CA certificate was received. Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to verify that this certificate is valid and has a lifetime at which includes the current timeframe.

Module
pki390.cpp

Procedure name
None.

EZD1039I IKE configuration file repeatable parameter pname was specified more than the allowed maximum of max_num times

Explanation
The IKE configuration file contains a repeatable parameter that was specified more times than is allowed. 
 pname is the repeatable parameter name.
 max_num is the maximum number of times this repeatable parameter is allowed to be specified.

System action
If the error occurs during IKE startup, IKE configuration file processing ends, and the IKE daemon ends. If the error occurs due to a MODIFY REFRESH command, IKE configuration file processing ends, but IKE remains active. IKE configuration retains all values prior to the MODIFY REFRESH command.

Operator response
Change the IKE configuration file so that pname is specified no more than the value of max_num. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file.

System programmer response
None.

Module
ike_config.cpp

Procedure name
None.

EZD1040I Phase phase Security Association retransmit timeout src IP : src_spec dest IP : dest_spec src port : src_port dest port : dest_port protocol : protocol

Explanation
The Internet Key Exchange (IKE) daemon exhausted the retransmit limit set for a single phase 1 or phase 2 message retransmission.
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*phase* is the phase (1 or 2) of this SA.

*src_spec* is the local security endpoint IP specification.

*dest_spec* is the remote security endpoint IP specification.

*src_port* is the source port.

*dest_port* is the destination port.

*protocol* is the protocol this SA used.

**System action**

The SA negotiation failed; IKE daemon processing continues.

**Operator response**

This might occur if there are network problems or the remote security endpoint is not responding. Take corrective action if necessary, and try the negotiation again. If failures continue, ensure that the remote security endpoint is responding.

**System programmer response**

None.

**Module**

retrans.cpp

**Procedure name**

None.

**EZD1041I** Error encountered when sending *ike_event* between src IP : *src_spec* dest IP : *dest_spec* src port : *src_port* dest port : *dest_port* protocol : *protocol* - errno | errnojr | description

**Explanation**

The Internet Key Exchange (IKE) daemon detected an error while transmitting or retransmitting a message to a remote security endpoint.

- When configured without the IBM Configuration Assistant for z/OS Communications Server, messages will be transmitted based on the IkeRetries and IkeInitWait parameters on the IkeConfig statement. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about the IkeConfig statement.

- When configured with the IBM Configuration Assistant for z/OS Communications Server, messages will be retransmitted based on the Advanced: Phase 1 IKE key negotiation retry tuning and Phase 2 IKE data negotiation retry tuning settings in the corresponding Image in the GUI. See the online helps in the GUI for additional information.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:

*ike_event* The type of message sent.
src_spec
The local security endpoint address specification.

dest_spec
The remote security endpoint address specification.

src_port
The source port.

dest_port
The destination port.

protocol
The protocol being used.

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

errnojr
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

description
Describes the meaning of the errno value.

System action
The message was not successfully transmitted; IKE daemon processing continues.

Operator response
This might occur if there are network problems or the remote security endpoint is not responding. Take corrective action if necessary, and try the negotiation again. If failures continue, ensure that the remote security endpoint is responding.

System programmer response
None.

Module
infoXchg.cpp, phase1.cpp, phaseII_sa.cpp, retrans.cpp, sa.cpp

Procedure name
None.

EZD1042I IKE CONFIGURATION FILE ERROR : MODIFY REFRESH COMMAND IS REJECTED - IKE DAEMON IS USING CONFIGURATION VALUES PRIOR TO COMMAND

Explanation
An error occurred while processing the IKE configuration file during a MODIFY REFRESH command. The MODIFY REFRESH command is rejected and the IKE daemon uses all the configuration values that were specified prior to the command.

System action
MODIFY REFRESH command is rejected.

Operator response
Correct the configuration file error and re-enter the MODIFY REFRESH command.
System programmer response
None.

Module
mdfysrvr.c

Procedure name
None.

EZD1043I Unable to open message catalog (catalog_name) errno | errnojr | description - default messages will be used

Explanation
The specified message catalog cannot be opened. This might occur if the catalog is missing, corrupted, or has incorrect permissions. This might also occur if the IKE daemon cannot find the catalog because environment variables or paths are set up incorrectly on the system. The NLSPATH environment variable is used to set the location of the message catalog. By default, this variable is set to /usr/lib/nls/msg/%L/%N.

catalog_name is the name of the catalog that the IKE daemon attempted to open.

erro is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errno) information in z/OS UNIX System Services Messages and Codes.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

description describes the meaning of errno.

System action
IKE daemon processing continues and will use internal default messages instead of messages from the external message catalog.

Operator response
If you want to use the external message catalog, correct the indicated error and restart the IKE daemon. If the default messages are acceptable, no action is necessary. See the information about the NLSPATH environment variable in z/OS UNIX System Services Programming Tools.

System programmer response
None.

Module
fwutil.c

Procedure name
None.

EZD1044I The ID (id_X.500_string) sent by the remote security endpoint in the ID payload does not match the subject name or any of the subject alternate names in the certificate used by the remote security endpoint to generate its signature.
Explanation
The IKE negotiation will probably fail because the identities in the ID payload and in the certificate do not match. This mismatch occurred during verification of the remote security endpoint identity while using digital signature mode authentication.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

$id_X.500_string$ is the X.500 string from the ID payload.

System action
The certificate cannot be used and the negotiation will probably fail; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint about this error and ask the administrator to ensure that they are using a certificate that matches the identity they are sending.

Module
pki390.cpp

Procedure name
None.

EZD1045I    IKE INITIALIZATION ERROR : error_data

Explanation
An error occurred while the Internet Key Exchange (IKE) daemon was initializing.

$error_data$ is an explanation of the error that occurred.

System action
The IKE daemon ends.

Operator response
Use the error explanation provided and other messages in syslog with the same message instance number to fix the error and restart the IKE daemon.

System programmer response
None.

Module
fw_initterm.c, ike_config.cpp

Procedure name
None.

EZD1046I    IKE INITIALIZATION COMPLETE
Explanation
The Internet Key Exchange (IKE) daemon successfully completed initialization.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

Module
main.cpp

Procedure name
None.

EZD1049I  IKE INITIALIZATION FAILURE

Explanation
The Internet Key Exchange (IKE) daemon failed to initialize.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

System action
The IKE daemon ends.

Operator response
Check for other messages in syslog with the same message instance number to indicate the reason for the failure. If the problem can be corrected, start the IKE daemon again. If the problem continues, notify the system programmer.

System programmer response
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

Module
fw_initterm.c

Procedure name
None.

EZD1051I  Phase 1 Security Association for DVIPA DVIPA_addr is not re-established with remote security endpoint ip_addr
**Explanation**
During a dynamic virtual IP address (DVIPA) takeover or giveback, the IKE daemon detected that the phase 1 Security Association (SA) for the specified DVIPA address was not re-established with its remote security endpoint. An error might have occurred during the phase 1 SA negotiation process.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*DVIPA_addr* is the dynamic virtual IP address.

*ip_addr* is the IP address of the remote security endpoint.

**System action**
The IKE phase 1 SA was not re-established for the DVIPA; IKE daemon processing continues.

**Operator response**
Check the logs for an entry indicating what the phase 1 SA negotiation error is.

**System programmer response**
None.

**Module**

*sa.cpp*

**Procedure name**

None.

---

**EZD1052I Phase 1 Security Association for DVIPA DVIPA_addr is re-established with remote security endpoint ip_addr**

**Explanation**
During a dynamic virtual IP address (DVIPA) takeover or giveback, the IKE daemon successfully re-established the phase 1 Security Association (SA) with the specified remote security endpoint.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*DVIPA_addr* is the dynamic virtual IP address.

*ip_addr* is the IP address of the remote security endpoint.

**System action**
The IKE phase 1 SA is re-established for the DVIPA.

**Operator response**
None.

**System programmer response**
None.
Module
phase1.cpp

Procedure name
None.

EZD1053I Discontiguous subnet mask mask is prohibited

Explanation
Discontiguous subnet masks, for example 255.0.0.255, are not permitted to be proposed in a Security Association (SA) negotiation.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

mask is the discontiguous subnet mask that is rejected by the IKE daemon.

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
Ensure that all IKE peers configure contiguous subnet masks for tunnel negotiations.

System programmer response
None.

Module
cfg_adapter.cpp

Procedure name
None.

EZD1054I Request to request could not be completed

Explanation
The Internet Key Exchange (IKE) daemon did not successfully complete the request specified.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

request is the request that could not be completed. request is one of the following:

- establish phase 1 Security Association
- establish phase 2 Security Association
- activate phase 2 Security Association
- refresh Security Association
- refresh phase 2 Security Association
- expire Security Association
- handle stack stop
- activate on-demand Security Association
• complete asynchronous verification request
• repopulate phase 1 Security Association
• complete DVIPA event
• display Security Association information
• activate auto-activate Security Association
• deactivate Security Association
• update policy

System action
The request failed; IKE daemon processing continues.

Operator response
See other messages that have the same message instance number to correct the problem, and try the operation again.

System programmer response
None.

Module
anchor_ureq.cpp

Procedure name
None.

EZD1055I IKE TERMINATION COMPLETE

Explanation
The Internet Key Exchange (IKE) daemon terminated successfully.

System action
The IKE daemon ends.

Operator response
None.

System programmer response
None.

Module
main.cpp

Procedure name
None.

EZD1057I Unsupported configuration: endpoint data endpoint address is specified as a type for a transport mode Security Association initiated mode
**Explanation**
A data endpoint was specified as an IP subnet or IP address range for a transport mode Security Association (SA). Only single IP address types are supported.

*endpoint* is either **local** or **remote**.

*address* is the base address value specified in the configuration policy.

*type* indicates whether the configuration specified subnet or range.

*mode* indicates whether the SA was initiated locally or remotely.

**System action**
The dynamic tunnel activation fails; IKE daemon processing continues.

**Operator response**
If you want a single SA to protect traffic for an IP subnet or IP address range, then configure a tunnel mode SA. If the message indicates that the SA was initiated locally, then change the local configuration. If the message indicates that the SA was initiated remotely, then the configuration on the remote system must be changed. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring dynamic tunnels.

**System programmer response**
None.

**Module**
policy.cpp

**Procedure name**
None.

**EZD1058I**  IKE STATUS FOR STACK *stackname* IS UP

**Explanation**
The Internet Key Exchange (IKE) daemon has detected a status change for a stack. The IKE daemon can establish security associations only for a stack with status UP. The IKE daemon might detect a stack status change for several reasons such as normal termination of the stack while the IKE daemon is running. *stackname* is the name of the stack that has changed status.

**System action**
IKE daemon processing continues.

**Operator response**
If the status change is unexpected, check the log for related messages from the IKE daemon or the stack.

**System programmer response**
None.

**Module**
stackobj.cpp
**EZD1059I  IKE CONNECTED TO PAGENT**

**Explanation**
The Internet Key Exchange (IKE) daemon connected to the Policy Agent (PAGENT).

**System action**
IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
policymgr.cpp

**Procedure name**
None.

**EZD1060I  IKE CONNECTION TO PAGENT FAILED**

**Explanation**
The Internet Key Exchange (IKE) daemon failed to connect to the Policy Agent (PAGENT).

**System action**
The IKE daemon ends.

**Operator response**
Start PAGENT before starting the IKE daemon.

**System programmer response**
None.

**Module**
policymgr.cpp

**Procedure name**
None.

**EZD1061I  IKE CONNECTING TO PAGENT**

**Explanation**
The Internet Key Exchange (IKE) daemon is connecting to the Policy Agent (PAGENT).
**System action**
The IKE daemon waits for the PAGENT connection to complete.

**Operator response**
None.

**System programmer response**
None.

**Module**
policymgr.cpp

**Procedure name**
None.

---

**EZD1062A  IKE RETRYING CONNECTION TO PAGENT [ FOR max_wait SECONDS]**

**Explanation**
The Internet Key Exchange (IKE) daemon is trying the connection to the Policy Agent (PAGENT) again.

`max_wait` is the maximum amount of time, in seconds, that the IKE daemon will continue to try connecting to PAGENT again. The absence of this information indicates that the IKE daemon will keep trying indefinitely.

**System action**
IKE daemon processing continues to try the connection to PAGENT again.

**Operator response**
Confirm that PAGENT is started.

**System programmer response**
None.

**Module**
policymgr.cpp

**Procedure name**
None.

---

**EZD1063I  IKE DISCONNECTING FROM PAGENT**

**Explanation**
The Internet Key Exchange (IKE) daemon is disconnecting from the Policy Agent (PAGENT).

**System action**
IKE daemon processing continues.

**Operator response**
None.
**System programmer response**
None.

**Module**
policymgr.cpp

**Procedure name**
None.

**EZD1064I MODIFIED COMMAND OPTION (opt) IS NOT SUPPORTED**

**Explanation**
The MODIFY command option opt is not a supported option for the IKE daemon.

opt is the option that is not supported.

**System action**
The IKE configuration file processing ended, but the IKE daemon remains active. The IKE configuration retains all values prior to the MODIFY command.

**Operator response**
See the information about the MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information the correct syntax of the MODIFY command.

**System programmer response**
None.

**Module**
mdfysrvr.c

**Procedure name**
None.

**EZD1065I IKE daemon could not establish a socket for port port for stack stackname**

**Explanation**
The Internet Key Exchange (IKE) daemon was unable to establish a socket for the specified port and stack. To negotiate security associations for an IPSECURITY stack, the IKE daemon must be able to establish sockets on ports 500 and 4500 for the stack.

port is the port for which the IKE daemon was trying to establish a socket. Possible values are 500 or 4500.

stackname is the name of the stack for which the socket could not be established.

**System action**
The IKE daemon is unable to negotiate security associations for the given stack.

**Operator response**
Check syslog for other messages that indicate errors that might prevent the IKE daemon from establishing the specified socket and contact the system programmer.
System programmer response
Ensure that ports 500 and 4500 are not reserved for use by another application.

Module
simple_net.cpp

Procedure name
None.

EZD1066I  IKE MODIFY COMMAND ACCEPTED

Explanation
The Internet Key Exchange (IKE) daemon started processing a MODIFY command.

System action
IKE daemon processing continues to process the MODIFY command.

Operator response
None.

System programmer response
None.

Module
mdfysrvr.c

Procedure name
None.

EZD1067I  IKE configuration file parameter ( pname ) is non-refreshable on line linenum

Explanation
The IKE configuration file parameter pname cannot be refreshed using the MODIFY command.
In the message text:

pname
  The parameter that cannot be refreshed.

linenum
  The line of the configuration file where the parameter was found.

System action
The IKE daemon configuration processing continues.

Operator response
None.

System programmer response
None.
User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp

Routing code
10

Descriptor code
12

Example
EZD1067I IKE configuration file parameter (KeyRing) is non-refreshable on line 5

EZD1068I IKE POLICY UPDATED FOR STACK stackname

Explanation
IKE policy lists were successfully updated for the specified stack. An EZD1070I message can be found in syslog for any of the IKE policy lists that were updated (up to 5 lists can updated).

stackname is the name of the stack that had a policy update.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

Module
policymgr.cpp

Procedure name
PMAdd_PolicyCacheForStack

EZD1069I IKE POLICY UPDATE FAILED FOR STACK stackname
Explanation
An IKE policy list failed to be successfully built, therefore none of the policy list changes will be performed for this stack. This message indicates that the IKE policy lists were not updated because a failure was detected when building an IKE policy list. All the policy lists that were successfully built will be deleted and this stack will use the prior policy information. Also, an EZD1071I message can be found in syslog for the IKE policy list that could not be built. There might be EZD1070I messages for each of the IKE policy lists that were successfully built (these lists are deleted when the build error is detected for an IKE policy list).

stackname is the name of the stack for which the policy update failed.

System action
IKE policy information will not be updated; IKE daemon processing continues.

Operator response
You can use the MODIFY REFRESH command to modify the PAGENT policy information, which will cause the IKE daemon to receive an asynchronous notification of the changes to policy information. Then verify that an EZD1068I message is reported on the console or in syslog to indicate that the IKE policy information was successfully updated for this stack. If the problem persists, contact the system programmer.

System programmer response
The failure might be the result of IKED not having RACF authority to access the policies. See Steps for authorizing the IKE daemon to RACF in z/OS Communications Server: IP Configuration Guide for more information. Otherwise, contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

Module
policymgr.cpp

Procedure name
PMConnect function or policyAsync exit

EZD1070I      IKE policy list policy-list-name updated for stack stackname

Explanation
An IKE policy list was successfully updated. This message will be displayed in syslog for all policy lists that the IKE daemon successfully updated. EZD1068I will be sent to the console and syslog when all lists were successfully updated. This message might be issued when a subsequent policy list fails to be updated (for which an EZD1071I message will be recorded), and an EZD1069I message will be sent to the console and to the syslog. In a failure case, all the previously successfully built policy lists indicated by the EZD1070I message will be deleted.

policy-list-name is the name of the IKE policy list and has one of the following values:

- KeyExchangeRule
- IpDynVpnAction
- IpLocalStartAction
- LocalDynVpnRule
- AutoActLocalDynVpnRule

stackname is the TCP/IP stack name for which this IKE policy is being updated.

System action
IKE daemon processing continues.
Explanation
An error was detected when attempting to build an IKE policy list. This message will be displayed in syslog for the policy lists for which the error was detected, followed by EZD1069I (sent to the console and syslog), to indicate that the IKE policy lists were not updated. All the previous IKE policy lists that were successfully built, which are indicated by an EZD1070I message, will be deleted.

`policy-list-name` is the name of the IKE policy list and has one of the following values:
- `KeyExchangeRule`
- `IpDynVpnAction`
- `IpLocalStartAction`
- `LocalDynVpnRule`
- `AutoActLocalDynVpnRule`

`stackname` is the stack name for which this IKE policy is being updated.

System action
IKE policy information is not updated; IKE daemon processing continues.

Operator response
Use the MODIFY REFRESH command to modify the PAGENT policy information, which will cause the IKE daemon to receive an asynchronous notification of the changes to policy information. Then verify that an EZD1068I message is reported on the console or syslog to indicate that the IKE policy information was successfully updated for this stack. If the problem persists, contact the system programmer.

System programmer response
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.
Unable to find Security Association by cookies (initiator_cookie / responder_cookie) between (local_ip / remote_ip)

Explanation
The Internet Key Exchange (IKE) daemon was unable to locate a Security Association (SA) in the SA table with the specified cookie pair.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

initiator_cookie is the 8-byte hexadecimal value of the initiator cookie.
responder_cookie is the 8-byte hexadecimal value of the responder cookie.
local_ip is the local IP address.
remote_ip is the remote IP address.

System action
The SA negotiation fails; IKE daemon processing continues.

Operator response
This error might be caused by receiving an incorrect cookie from the remote security endpoint when the IKE daemon is recycled during an SA negotiation or when additional messages are received after an SA is deactivated. If none of these actions happened, contact the system programmer.

System programmer response
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

Module
anchor_msg.cpp

Procedure name
None.

Error reading ipsec command request: errno errno (description) errnojr errnojr

Explanation
The Internet Key Exchange (IKE) daemon detected an error reading an ipsec command request.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

description describes the meaning of errno.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The command request ends; IKE daemon processing continues.
Operator response
This error might be caused by a temporary restraint on resources; try the command again.

System programmer response
None.

Module
simple_cmd_req.cpp

Procedure name
None.

EZD1074I Error writing ipsec command response: errno errno (description) errnojerrnoj

Explanation
The Internet Key Exchange (IKE) daemon detected an error writing an ipsec command response.
errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.
description describes the meaning of errno.
errnoj is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The command request ends; IKE daemon processing continues.

Operator response
This error might be caused by a temporary restraint on resources; try the command again.

System programmer response
None.

Module
simple_cmd_req.cpp

Procedure name
None.

EZD1075I Received ISAKMP error notification message: err_msg_type

Explanation
The Internet Key Exchange (IKE) daemon received an ISAKMP error notification message. This error indicates that a Security Association (SA) negotiation failure occurred.
err_msg_type is the type of ISAKMP error notification received.

System action
The SA negotiation failed; IKE daemon processing continues.
Operator response
Use the err_msg_type value to identify the failure and verify that the security policy is specified correctly.
See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

System programmer response
None.

Module
infoXchg.cpp

Procedure name
None.

EZD1076I The function function returned error code ec while processing distinguished name [dn]

Explanation
An error was encountered while processing a distinguished name (DN).
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

function is the name of the function that returned an error.

ec is the error code from the function specified. For a list of common error codes returned by these functions, see the z/OS Cryptographic Services System SSL Programming.

dn is the distinguished name that is being processed if the function returning the error was gsk_dn_to_name().

System action
The SA negotiation failed; IKE daemon processing continues.

Operator response
These errors are usually caused by incorrectly formed DNs. See RFC 2253 for more information about forming DNs and check the DNs in your policy configuration file for incorrect characters (such as surrounding double quotes) and try the operation again. See Appendix A, “Related protocol specifications,” on page 1365 for directions on how to get a copy of the RFC.

System programmer response
None.

Module
fwconvert.cpp

Procedure name
None.

EZD1077I Could not find the IpDynVpnAction named name

Explanation
An IKE negotiation failed because the IKE daemon was unable to find the IpDynVpnAction specified.
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

`name` is the IpDynVpnAction that could not be found.

**System action**
The Security Association negotiation fails; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

**Module**
policymgr.cpp

**Procedure name**
None.

---

**EZD1079I** IKE POLICY PURGED FOR STACK `stackname`

**Explanation**
IKE policy lists have been successfully purged for the specified stack.

`stackname` is the name of the stack that had a policy purge.

**System action**
IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
policymgr.cpp

**Procedure name**
None.

---

**EZD1081I** Could not find the IpLocalStartAction named `name`

**Explanation**
An IKE negotiation failed because the IKE daemon was unable to find the IpLocalStartAction specified.
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

`name` is the `IpLocalStartAction` that could not be found.

**System action**
The Security Association negotiation fails; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Contact IBM software support services and provide a syslog that includes this message. If available, provide a CTRACE for component SYSTCPIK.

**Module**
policymgr.cpp

**Procedure name**
None.

**EZD1082I** Could not find the `LocalDynVpnRule` named `name`

**Explanation**
An IKE negotiation failed because the IKE daemon was unable to find the `LocalDynVpnRule` specified.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

`name` is the `LocalDynVpnRule` that could not be found.

**System action**
The Security Association negotiation fails; IKE daemon processing continues.

**Operator response**
Ensure that the `LocalDynVpnRule` name matches a configured `LocalDynVpnRule`. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about the `LocalDynVpnRule` statement.

**System programmer response**
None.

**Module**
policymgr.cpp

**Procedure name**
None.

**EZD1083I** Local policy (`p1_action_name`) does not allow local initiation of a phase 1 Security Association negotiation
**Explanation**
The local IKE daemon is attempting to initiate a phase 1 Security Association (SA) negotiation, but the local policy specifies that it cannot be the initiator.

*p1_action_name* is the name of the action.

- In the policy agent configuration file, *p1_action_name* is the name of the KeyExchangeAction statement associated with this negotiation.
- When configured with the IBM Configuration Assistant for z/OS Communications Server, *p1_action_name* corresponds to the name of a Connectivity Rule in the GUI. *p1_action_name* also contains a numeric suffix appended to the Connectivity Rule name to guarantee uniqueness.

**System action**
The SA negotiation failed; IKE daemon processing continues.

**Operator response**
Check the IKE daemon's initiation role for this SA.

When configured without the IBM Configuration Assistant for z/OS Communications Server, if the local IKE daemon should be able to initiate the negotiation for this SA, then change the daemon's HowToInitiate parameter in the appropriate KeyExchangeAction statement to **main** or **aggressive**. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

When configured with the IBM Configuration Assistant for z/OS Communications Server, if the local IKE daemon should be able to initiate the negotiation for this SA, then edit the corresponding Connectivity Rule GUI and change the Initiator mode setting on the Advanced IPSec: Dynamic Tunnels: Key Exchange Settings panel to either **Main** or **Aggressive**. See the online helps in the GUI for additional information.

**System programmer response**
None.

**Module**
policy.cpp

**Procedure name**
None.

**EZD1085I**
A message was discarded because it was received from a remote peer behind an NAPT - src IP : sourceIP src port : sourceport dest IP : destIP dest port : destport

**Explanation**
The Internet Key Exchange (IKE) daemon is not permitted to allow a Security Association (SA) with a remote peer behind a network address port translation (NAPT). Therefore, any IKE message received from a remote peer with a source port other than 500 or 4500 will be discarded.

*sourceIP* is the source IP address.

*sourceport* is the source port.

*destIP* is the destination IP address.

*destport* is the destination port.

**System action**
The IKE message from the remote peer will be discarded; the IKE daemon continues.
Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that the IKE daemon cannot accept messages from a peer behind an NAPT.

Module
simple_net.cpp

Procedure name
None.

EZD1086I Negotiating a phase 1 Security Association due to a remote security endpoint IP address change - original IP address / port : origip / origport changed IP address / port : changedip / changedport

Explanation
A new phase 1 Security Association (SA) is being negotiated because an IP address change was detected for a remote security endpoint. This occurrence is usually related to a re-boot of the NAT device. If this new phase 1 negotiation completes successfully, then all of the previous SAs with the remote security endpoint are deleted. If this negotiation successfully completes, message EZD1087I will be issued, or if the negotiation fails, message EZD1102I will be issued.

origip is the IP address of the established SA.
origport is the port of the established SA.
changedip is the new IP address.
changedport is the new port.

System action
A new phase 1 SA is negotiating; IKE daemon processing continues.

Operator response
None.

System programmer response
None.

Module
sa.cpp

Procedure name
None.

EZD1087I Negotiation of a phase 1 Security Association due to a remote security endpoint IP address change succeeded - original IP address / port : origip / origport new IP address / port : newip / newport
Explanation
This message indicates that the phase 1 Security Association (SA) being negotiated in response to a remote security endpoint IP address change succeeded. This message is associated with a previously issued EZD1086I message.

origip is the IP address of the previously established SA.
origport is the port of the previously established SA.
newip is the new IP address.
newport is the new port.

System action
Negotiation of the phase 1 SA succeeds; IKE daemon processing continues.

Operator response
Contact the system programmer if this message is seen often.

System programmer response
If this message is seen often, determine the cause of the IP address change. It might be caused by a reboot of the NAT device in front of the remote peer or by an expired NAT mapping. Notify the administrator of the remote security endpoint about the error. If the change is caused by an expired NAT mapping, the administrator of the remote security endpoint should take action to prevent future NAT mapping expirations.

Module
oakley_kep.cpp

Procedure name
None.

EZD1088I IKE received an unsupported identity address type (idtype - idtypestr) for a Security Association traversing a NAT

Explanation
An identity address must be specified as a single IPv4 address.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:

identity
Possible values are:

IDci
The identity address type of the initiator.

IDcr
The identity address type of the responder.

NAT-OA
The NAT original address.

idtype
The number of the identity address type that is not supported. These values are defined in RFC2407. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.
idtypestr
The identity address type that is not supported. If the identity address type is not known, the value of idtypestr is Unknown.

System action
The phase 2 SA negotiation fails; IKE daemon processing continues.

System programmer response
Ensure that only single IPv4 addresses are specified as data endpoints when traversing a NAT. Notify the administrator of the remote security endpoint and ask the administrator to ensure that only single IPv4 addresses are specified as data endpoints when traversing a NAT.

User response
Contact the system programmer.

Module
oakley_phaseII.cpp

Example
None.

EZD1089I  A tunnel mode Security Association traversing a NAT does not have its local IPSec traffic endpoint residing on this node

Explanation
During the negotiation of a tunnel mode Security Association (SA), it was determined that the local IPSec traffic endpoint did not end on this z/OS node. z/OS is providing NAT Traversal support for a defined group of configurations where z/OS is running the IKE daemon. See the information about IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations.

System action
The tunnel mode SA negotiation fails; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Alter the local policy configuration so that the local IPSec traffic endpoint is local to this z/OS.

When configured without the IBM Configuration Assistant for z/OS Communications Server, in the policy agent configuration file, this IP address is the IpSourceAddr parameter on the IpFilterRule. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

When configured with the IBM Configuration Assistant for z/OS Communications Server, edit the corresponding Connectivity Rule in the GUI and ensure the local data endpoint address is one that is local to the TCP/IP stack. Gateway-to-host and gateway-to-gateway topologies are not supported for NAT. See the online helps in the GUI for additional information.

Module
oakley_phaseII.cpp
**EZD1090I**  
Initiation of a phase 2 Security Association negotiation for a new dynamic tunnel failed because the remote security endpoint is a security gateway

**Explanation**
When traversing a NAT, a local initiation of a phase 2 Security Association (SA) for a new dynamic tunnel with a remote security endpoint that is a security gateway is not supported. To use this configuration, the remote security endpoint must be the initiator. z/OS is providing NAT traversal support for a defined group of configurations where z/OS is running the IKE daemon. See the information about IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations.

**System action**
The phase 2 SA negotiation fails; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Notify the administrator of the remote security endpoint that it must be the initiator when using this configuration.

**Module**
sa.cpp

**EZD1092I**  
Protocol error encountered during phase phase message processing rsn=rsn - message discarded

**Explanation**
A protocol error occurred during IKE message processing. The rsn field provides more information about the received message.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

phase is 1 or 2 indicating the phase of negotiation when the error occurred.

rsn is the reason code that provides additional information about the received message. Possible values are:

1. The first payload in the quick mode (phase 2) message was not a hash payload.
2. The second payload in message 1 or 2 of a quick mode (phase 2) exchange was not a Security Association payload.
3. A quick mode (phase 2) message was not encrypted.
4. The received message contained unexpected payloads or was missing payloads that are required by RFC 2409 (The Internet Key Exchange).
The received message did not contain the required number of NAT-OA payloads.

The received message contained too many NAT-OA payloads.

The received message utilized an unexpected port.

The message length indicated in the ISAKMP header of the message is too large.

The received message is missing a required key exchange, NONCE payload, or both.

The received message is missing a required ID payload.

The received message is missing a required hash or signature payload.

The received message contains Diffie-Hellman information that is too long.

The received message did not contain an expected certificate payload.

The received message is too short to be a valid ISAKMP message.

The received message is too large to buffer.

The received message contains a next payload field that is unrecognized.

The received message does not contain a valid ISAKMP major and minor version.

The received message's exchange type is not supported.

The received message contains no payloads.

The received message contains a payload that is shorter than the reported size.

The received message contains a payload that is longer than the reported size.

The received message contains a payload with no data.

The received message contains a payload that is not the correct payload size.

The received message contains an incorrect SPI size.

The received message contains non-zero data in a field that must be set to 0.

The received message contains an unsupported Domain Of Interpretation (doi) value.

The received message contains an unsupported situation value.

The received message contains an unsupported protocol value.

The received message contains an unsupported ID type value.
The received message contains an unsupported certificate type value.

The received phase 1 message 1 contains encrypted data.

The received message contains an SA payload without a required hash payload.

The received message contains non-SA payloads before the first SA payload.

The received message does not contain a proposal payload in the required order.

The received message does not contain a transform payload in the required order.

The received message contains an incorrect size for the ID type received.

System action
The SA negotiation fails; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that a protocol error has occurred.

Module
oakely_phaseII.cpp

Procedure name
None.

EZD1093I  Policy mismatch: statement (state_num) requires parameter (parameter) with value (policy_val) but proposal (prop_num) value is (prop_val)

Explanation
The Internet Key Exchange (IKE) daemon was unable to accept a proposal because there was a mismatch in the configured policy. IKE daemon processing continues to the next proposal. If no proposals are accepted, the Security Association negotiation will fail. This will be indicated by an EZD0985I, EZD1021I, or EZD1022I message later in syslog.

statement indicates whether the mismatched parameter was configured on the KeyExchangeOffer, IpDataOffer, or IpDynVpnAction statement in the policy configuration file.

When configured with the IBM Configuration Assistant for z/OS Communications Server:

- The KeyExchangeOffer statements are located on the corresponding Connectivity Rule's Advanced IPSec: Dynamic Tunnels: Key Exchange Settings panel. Use the KeyExchangeRule name from message EZD1021I to identify the Connectivity Rule.
- The IpDataOffer statement is located in the corresponding security level. However, if the parameter is HowToEncap, this setting is located on the Connectivity Rule's Advanced IPSec: Dynamic Tunnels: OnDemand Granularity / Encapsulation Mode panel. Use the DynVpnAction name from message EZD1022I to identify the security level. Use the IpFilerRule name from message EZD1022I to identify the Connectivity Rule.
• The IpDynVpnAction statement corresponds to the security level in the GUI. Use the DynVpnAction name from message EZD1022I to identify the corresponding security level.

state_num is the number of a statement referenced from the policy. The number corresponds to the order of the references in the policy. Therefore, the first statement referenced from the policy would have number 1 in this message.

parameter is the parameter that encountered a mismatch. If parameter is ExtendedSequenceNumbers, this value is not configurable in z/OS policy. For all other values of parameter, see the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about the parameter specified, or "No" if the parameter is ExtendedSequenceNumbers.

policy_val is the value that is configured in the policy.

prop_num is the number of the proposal that is being compared. The number corresponds to the order of proposals in an offer received. Therefore, the first proposal received in an offer would have number 1 in this message.

prop_val is the value contained in the proposal for this parameter.

System action
If the IKE daemon does not accept any of the proposals, the negotiation fails; IKE daemon processing continues.

Operator response
If the proposal that contains the mismatch is the one that should be accepted, either alter the local policy to accept the value in this proposal or notify the administrator of the remote security endpoint about the mismatch and ask the administrator to alter the remote configuration to propose the correct values.

See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

System programmer response
None.

Module
policy.cpp

Procedure name
None.

EZD1094I IKE STATUS FOR STACK stackname IS DOWN

Explanation
The Internet Key Exchange (IKE) daemon detected a status change for a stack. The IKE daemon can establish Security Associations only for a stack with status UP. The IKE daemon might detect a stack status change for several reasons such as normal termination of the stack while the IKE daemon is running.

stackname is the name of the stack that has changed status.

System action
IKE daemon processing continues.

Operator response
If the status change is unexpected, check the log for related messages from the IKE daemon or the stack.
System programmer response
None.

Module
stackobj.cpp

Procedure name
None.

EZD1095I Unsupported configuration: location data endpoint dataaddr does not match security endpoint endptaddr for a transport mode Security Association initiated method

Explanation
The IP address for the data endpoint must be the same as the IP address for the security endpoint when using a transport mode Security Association (SA).

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

location describes the data endpoint. The possible values are local and remote.

dataaddr is the data endpoint IP address.

endptaddr is the security endpoint IP address.

method describes how the SA was initiated. The possible values are locally and remotely.

System action
The SA negotiation fails; IKE daemon processing continues.

Operator response
If the data endpoint and security endpoint are different, use a tunnel mode SA; otherwise change the configuration so that the data endpoint and security endpoint addresses match. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

System programmer response
None.

Module
policy.cpp

Procedure name
None.

EZD1096I Local IP address addr is not available

Explanation
The source address used for a Security Association (SA) negotiation must be an address owned by the stack. If the local address is a distributed DVIPA, then the stack must be acting as the distributor.

addr is the source address for the attempted negotiation.
**System action**
The SA negotiation fails; IKE daemon processing continues.

**Operator response**
If the SA negotiation was locally activated, verify that the IP address is local to this stack. Use the Netstat HOME/-h command to view the list of IP addresses local to this stack. See the information about the Netstat HOME/-h in z/OS Communications Server: IP System Administrator's Commands for information about displaying the Netstat HOME/-h report.

When configured without the IBM Configuration Assistant for z/OS Communications Server, check for LocalDynVpnRule statements with LocalIp or LocalIpRef parameters that specify an IP address that is not local to this stack and, if necessary, correct the policy. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

When configured with the IBM Configuration Assistant for z/OS Communications Server, edit the corresponding Connectivity Rule in the GUI and verify that the local IP address is local to this stack. If your local data endpoint is a single IP address, verify that it is local to this stack. If the local data endpoint value is an asterisk (*), verify that your Advanced IPSec: Dynamic Tunnels: Local Activation Settings are correct and contain only IP addresses local to this stack. See the online helps in the GUI for additional information.

**System programmer response**
None.

**Module**
anchor_ureq.cpp

**Procedure name**
None.

**EZD1097I**  Phase 2 Security Association for DVIPA dvipa_addr is re-established with remote security endpoint ip_addr

**Explanation**
During a DVIPA takeover or giveback, the IKE daemon successfully re-established the phase 2 Security Association (SA) with the specified remote security endpoint.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*dvipa_addr* is the dynamic virtual IP address.

*ip_addr* is the IP address of the remote security endpoint.

**System action**
The IKE phase 2 SA is re-established for the DVIPA; IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
None.
**EZD1098I**  Protocol error encountered during phase *phase* message processing rsn=*rsn* - negotiation continues

**Explanation**
A protocol error occurred during IKE message processing. The *rsn* field provides more information about the received message.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

*phase* is 1 or 2 indicating the phase of the negotiation when the error occurred.

*rsn* is the reason code that provides additional information about the received message. Possible values are:

1. A NAT-D payload received contained an invalid hash length. NAT detection cannot be performed.
2. The use of NAT traversal was agreed to but the remote security endpoint did not send the minimum number of NAT-D payloads required.

**System action**
The SA negotiation continues; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Notify the administrator of the remote security endpoint that a protocol error has occurred.

---

**EZD1099I**  A message generated dump has been created titled: *title*

**Explanation**
An IKE daemon message generated an address space dump. The message that generated the dump appears immediately after this message in the system log.

*title* is the text associated with the dump. The title contains the message number and associated message text that caused the dump to be generated.

**System action**
After the dump is created, IKE daemon processing continues processing.
**Operator response**
Contact the system programmer.

**System programmer response**
Capture the system log and the generated dump. Contact IBM software support services to analyze this data.

**Module**
fwutil.c

**Procedure name**
None.

**EZD1100I**  
A message generated dump was suppressed for message: message_number

**Explanation**
An IKE daemon message attempted to generate an address space dump. However, no more than two message-generated dumps can be created in a 15 minute period. Due to this criteria, the dump was suppressed. The message that attempted to generate the dump appears immediately after this message in the system log. *message_number* is the message number of the message that attempted to generate the dump.

**System action**
IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Capture the system log and any message-generated dumps that were created earlier. Contact IBM software support services to analyze this data.

**Module**
fwutil.c

**Procedure name**
None.

**EZD1101I**  
NAT detected and no valid IpDataOffers found

**Explanation**
This message is issued when no valid IpDataOffers are found during a negotiation traversing a NAT. When a NAT is being traversed, all IpDataOffers utilizing the AH protocol are ignored. To negotiate a security association (SA) in a NAT traversal environment, at least one Data Offer that does not contain authentication with AH must be configured.

**System action**
The SA negotiation fails; IKE daemon processing continues.
Operator response
Contact the system programmer.

System programmer response
Ensure that at least one data offer does not contain authentication with AH.

When configured without the IBM Configuration Assistant for z/OS Communications Server, ensure that at least one IpDataOffer has ESP or DoNot configured on the HowToAuth parameter in the configuration policy. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

When configured with the IBM Configuration Assistant for z/OS Communications Server, ensure that the security level in the GUI contains at least one Data Offer that either does not use Authentication or uses Authentication with the ESP authentication protocol. See the online helps in the GUI for additional information.

Module
policy.cpp

Procedure name
None.

EZD1102I Negotiation of a phase 1 Security Association due to a remote security endpoint IP address change failed - original IP address / port : origip / origport failed IP address / port : failedip / failedport

Explanation
This message indicates that a phase 1 Security Association (SA) being negotiated in response to a detected remote security endpoint IP address change failed. The changed IP address (failedip) that caused the negotiation might be caused by a reboot of the NAT device in front of the remote security endpoint or an expired NAT mapping. The changed IP address (failedip) might also be caused by a packet with an address that is not valid. This message is associated with a previously issued EZD1086I message.

origip is the IP address of the established SA.
origport is the port of the established SA.
failedip is the IP address of the failed negotiation.
failedport is the port of the failed negotiation.

System action
Negotiation of the phase 1 SA fails; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Check other messages in the system log with this IP address to determine the cause of the failure. If necessary, notify the administrator of the remote security endpoint about this error.

Module
anchor_ureq.cpp, oakley_kep.cpp, oakley_phaseII.cpp, sa.cpp
**EZD1103I  Informational exchange ignored because phase 1 Security Association is still being negotiated**

**Explanation**
This message is issued when an informational exchange message is received over a phase 1 Security Association that is still in the process of negotiation.

**System action**
The informational exchange message is discarded; IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
infoxchg.cpp

**EZD1104I  IKE detected a NAT while initiating a new dynamic tunnel using only tunnel mode IpDataOffers with a non-z/OS peer**

**Explanation**
The Internet Key Exchange (IKE) daemon is initiating a tunnel-mode Security Association (SA) for a new dynamic tunnel with a non-z/OS peer. The SA traverses a NAT. There might be problems with interoperability with the non-z/OS peer for a tunnel-mode SA. z/OS is providing NAT Traversal support for a defined group of configurations where z/OS is running the IKE daemon. See the information about IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations and interoperability considerations.

**System action**
The SA negotiation continues.

**Operator response**
If the SA negotiation fails or if data cannot be successfully sent over the SA, contact the system programmer.

**System programmer response**
Determine whether there is an interoperability concern that caused the SA negotiation or data to fail. See the information about IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations and interoperability considerations.

A possible solution is to use transport-mode IpDataOffers. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.
EZD1105I IKE detected a NAT while initiating a new dynamic tunnel using both tunnel and transport mode IpDataOffers with a non-z/OS peer

Explanation

The Internet Key Exchange (IKE) daemon is attempting to initiate a phase 2 Security Association (SA) for a new dynamic tunnel with a non-z/OS peer. The SA traverses a NAT. Both tunnel-mode and transport-mode IpDataOffers are proposed. If the peer selects a tunnel-mode proposal, interoperability issues might exist with the non-z/OS peer. z/OS is providing NAT Traversal support for a defined group of configurations where z/OS is running the IKE daemon. See the information about IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations and interoperability considerations.

System action

The SA negotiation continues.

Operator response

If the SA negotiation fails or if data cannot be successfully sent over the SA, contact the system programmer.

System programmer response

If the SA negotiation fails or if data cannot be successfully sent over the SA, see the information about IP security in z/OS Communications Server: IP Configuration Guide to determine if there is an interoperability concern that caused the SA negotiation or data to fail. Contact the remote peer's administrator to understand any interoperability considerations for the non-z/OS platform.

EZD1106I Tentative KeyExchangeRule (tentativeKER) and final KeyExchangeRule (finalKER) preshared keys do not match

Explanation

When using HowToAuthPeers PresharedKey, the SharedKey parameters on the tentative and final KeyExchangeRule statements must match. The initial KeyExchangeRule statement is found by matching on local and remote IP addresses, while the final match also takes the remote ID into account.

In the message text:

- **tentativeKER**
  - The initial KeyExchangeRule statement that was found by matching only IP addresses

- **finalKER**
  - The final KeyExchangeRule statement that was found by matching IP addresses and the remote ID

System action

The SA negotiation fails; IKE daemon processing continues.
Operator response
Contact the system programmer.

System programmer response
Change the policy configuration so the final rule will be matched tentatively and finally. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

Module
doi.cpp

Procedure name
None.

EZD1107I IKE detected a NAT while initiating a narrow Security Association negotiation for a new dynamic tunnel with a non-z/OS peer

Explanation
The Internet Key Exchange (IKE) daemon is attempting to initiate a narrow phase 2 Security Association (SA) for a new dynamic tunnel with a non-z/OS peer. The SA traverses a NAT. A narrow SA is an SA negotiated for specific ports, protocol, or both. Interoperability issues might exist with the non-z/OS peer when z/OS initiates a narrow phase 2 SA. z/OS is providing NAT Traversal support for a defined group of configurations where z/OS is running the IKE daemon. See the information about IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations.

System action
The SA negotiation continues.

Operator response
If the SA negotiation fails, contact the system programmer.

System programmer response
If the SA negotiation fails, see the information about IP security in z/OS Communications Server: IP Configuration Guide to determine if there is an interoperability concern that caused the phase 2 SA negotiation to fail. Contact the remote peer’s administrator to understand any interoperability considerations for the non-z/OS platform.

Module
oakley_phaseII.cpp

Procedure name
None.

EZD1108I Unable to retransmit message - associated phase 1 security association has expired

Explanation
The Internet Key Exchange (IKE) daemon attempted to retransmit a message using a phase 1 security association (SA) that is expired. This might occur in a NAT environment when the IP address of the remote security endpoint is remapped.
Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages.

**System action**
The message was not retransmitted and the SA negotiation failed; the IKE daemon continues.

**Operator response**
Verify that the required tunnels are activated, and if necessary, activate or refresh any required tunnels using the `ipsec` command. See the information about the managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

**System programmer response**
None.

**Module**
retrans.cpp

**Procedure name**
None.

---

EZD1109I  ICSF service CSNBSYD failed for AES decryption : return code = rc reason code = rsn

**Explanation**
The Integrated Cryptographic Service Facility (ICSF) service returned an error when called to perform AES decryption.

In the message text:

- **rc**
  The hexadecimal return code returned from the ICSF function call.

- **rsn**
  The hexadecimal reason code returned from the ICSF function call.

**System action**
The operation being performed fails; IKE daemon processing continues.

**Operator response**
Verify that ICSF is running. See the information about the ICSF and TSS Return and Reason Codes in z/OS Cryptographic Services ICSF Application Programmer's Guide for the meaning of the rc and rsn values and for specific actions to be taken.

**System programmer response**
None.

**User response**
Not applicable.
Problem determination
Not applicable.

Module
aes_obj.cpp

Example
None.

EZD1110I An IPv6 address was configured in an IP Security policy file for stackname that was not configured with IPCONFIG6 IPSECURITY

Explanation
The Internet Key Exchange (IKE) daemon attempted to retrieve an IpFilterRule from the specified stack. An IPv6 address was configured in an IP Security policy file, but this stack does not have IPCONFIG6 IPSECURITY configured.

In the message text:

stackname
The name of the TCP/IP stack.

System action
The request to retrieve the IpFilterRule fails; the IKE daemon continues.

Operator response
Contact the system programmer to determine whether the specified stack is intended to have IPCONFIG6 IPSECURITY configured.

System programmer response
If you want IPCONFIG6 IPSECURITY to be enabled for the specified stack, update the TCP/IP profile accordingly. Otherwise, correct the IP Security Policy file.

See the information about the IPCONFIG6 statement in z/OS Communications Server: IP Configuration Reference for more information about specifying IP security for a stack.

See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IPSec

Module
stackObj.cpp
Routing code  
*  

Descriptor code  
*  

Example  
Not applicable.

**EZD1111I**  
**KeyExchangeRule** *rulename* cannot specify a multicast address

**Explanation**  
An IKE negotiation using the KeyExchangeRule rule specified by the *rulename* value cannot continue because one or both of the addresses defined in the KeyExchangeRule rule is a multicast address. The security endpoints for an IKE negotiation can be specified as unicast addresses or they can use a restricted wildcard specification.

In the message text:

*rulename*  
The name of the KeyExchangeRule rule involved in the negotiation.

**System action**  
The IKE negotiation fails; the IKE daemon continues.

**Operator response**  
None.

**System programmer response**  
Change the addresses specified by the KeyExchangeRule rule in the local policy. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**User response**  
Not applicable.

**Problem determination**  
Not applicable.

**Source**  
z/OS Communication Server TCP/IP other application

**Module**  
policymgr.cpp

**Example**  
None.

**EZD1112I**  
**IpFilterRule** *rulename* cannot specify a multicast address
Explanation
An IKE negotiation using the IpFilterRule rule specified by the *rulename* value cannot continue because one or both of the addresses defined in the IpFilterRule rule is a multicast address. The data endpoints for a dynamic tunnel can be specified as unicast addresses or they can use a restricted wildcard specification.

In the message text:

`rulename`
The name of the IpFilterRule rule involved in the negotiation.

System action
The IKE negotiation fails; the IKE daemon continues.

Operator response
None.

System programmer response
Change the addresses specified by the IpFilterRule rule in the local policy. See the information about the Policy Agent and policy applications in *z/OS Communications Server: IP Configuration Reference* for more information about configuring policy.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communication Server TCP/IP other application

Module
policymgr.cpp

Example
None.

**EZD1113I** A delete message is being sent with LocalIp ANY for phase 1 security association *p1sa_id* (LocalIp : *local_ip* RemoteIp : *remote_ip*)

Explanation
A delete message is being sent to the IKE peer to terminate the phase 1 Security Association identified by the *p1sa_id* value between the local IP address specified by the *local_ip* value and the remote IP address specified by the *remote_ip* value. However, the local stack has already deleted its end of the tunnel. To insure that the delete message can be sent, the *local_ip* value is set to the value ANY (INADDR_ANY for IPv4 or IN6ADDR_ANY for IPv6).

In the message text:

*p1sa_id*
The phase 1 SA ID.

*local_ip*
The local IP address.
**remote_ip**
The remote IP address.

**System action**
The IKE daemon continues.

**Operator response**
With a source address value ANY, it is possible that the IKE peer might not handle the delete message. Some IKE implementations might require the source IP address as part of the tunnel identification process. If this message was received after the manual deletion of a DVIPA address, then use the **ipsec** command to deactivate tunnels that involve DVIPA addresses prior to manually deleting the DVIPA.

See the information about the managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the **ipsec** command syntax and options.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP other application

**Module**
infoXchg.cpp

**Example**
Not applicable.

**EZD1114I**  
*Policy mismatch: statement state_num requires parameter parameter that is not supported by proposal prop_num*

**Explanation**
The Internet Key Exchange (IKE) daemon was unable to accept a proposal because there was a mismatch in the configured policy. The IKE daemon continues to the next proposal. If no proposals are accepted, the Security Association negotiation will fail. This will be indicated by an EZD0985I, EZD1021I, or EZD1022I message later in syslog.

In the message text:

**statement**
Indicates whether the mismatched parameter was configured on the KeyExchangeOffer, IpDataOffer, or IpDynVpnAction statement in the policy configuration file.

When configured with the IBM Configuration Assistant for z/OS Communications Server:

- The KeyExchangeOffer statements are located on the corresponding connectivity rule's Advanced IPSec: Dynamic Tunnels: Key Exchange Settings panel. Use the KeyExchangeRule name from message EZD1021I to identify the connectivity rule. The IpDataOffer statement is located in the corresponding security level.
However, if the parameter is HowToEncap, this setting is located on the connectivity rule’s Advanced IPSec: Dynamic Tunnels: Key Exchange Settings panel. Use the DynVpnAction name from message EZD1022I to identify the security level. Use the IpFilerRule name from message EZD1022I to identify the connectivity rule.

- The IpDynVpnAction statement corresponds to the security level in the IBM Configuration Assistant for z/OS Communications Server. Use the DynVpnAction name from message EZD1022I to identify the corresponding security level.

**state_num**

The number of a statement referenced from the policy. The number corresponds to the order of the references in the policy. Therefore, the first statement referenced from the policy would have number 1 in this message.

**parameter**

The parameter that encountered a mismatch. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about the parameter specified.

**prop_num**

The number of the proposal that is being compared. The number corresponds to the order of proposals in an offer received.

**System action**

If the IKE daemon does not accept any of the proposals, the negotiation fails; the IKE daemon continues.

**System programmer response**

If the proposal that contains the mismatch is the one that should be accepted, either alter the local policy to accept the value in this proposal or notify the administrator of the remote security endpoint about the mismatch and ask the administrator to alter the remote configuration to propose the correct values. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**User response**

Contact the system programmer.

**Module**

policy.cpp

**Procedure name**

None.

**EZD1115I**

Policy mismatch: Proposal prop_num requires parameter parameter that is not supported by statement state_num

**Explanation**

The Internet Key Exchange (IKE) daemon was unable to accept a proposal because there was a mismatch in the configured policy. The IKE daemon continues to the next proposal. If no proposals are accepted, the Security Association negotiation will fail. This will be indicated by an EZD0985I, EZD1021I, or EZD1022I message later in syslog.

In the message text:

**prop_num**

The number of the proposal that is being compared. The number corresponds to the order of proposals in an offer received.
**parameter**

The parameter that encountered a mismatch. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about the parameter specified.

**statement**

Indicates whether the mismatched parameter was configured on the KeyExchangeOffer, IpDataOffer, or IpDynVpnAction statement in the policy configuration file.

When configured with the IBM Configuration Assistant for z/OS Communications Server:

- The KeyExchangeOffer statements are located on the corresponding connectivity rule's Advanced IPSec: Dynamic Tunnels: Key Exchange Settings panel. Use the KeyExchangeRule name from message EZD1021I to identify the connectivity rule. The IpDataOffer statement is located in the corresponding security level. However, if the parameter is HowToEncap, this setting is located on the connectivity rule's Advanced IPSec: Dynamic Tunnels: Key Exchange Settings panel. Use the DynVpnAction name from message EZD1022I to identify the security level. Use the IpFilerRule name from message EZD1022I to identify the connectivity rule.

- The IpDynVpnAction statement corresponds to the security level in the IBM Configuration Assistant for z/OS Communications Server. Use the DynVpnAction name from message EZD1022I to identify the corresponding security level.

**state_num**

The number of a statement referenced from the policy. The number corresponds to the order of the references in the policy. Therefore, the first statement referenced from the policy would have number 1 in this message.

**System action**

If the IKE daemon does not accept any of the proposals, the negotiation fails; the IKE daemon continues.

**System programmer response**

If the proposal that contains the mismatch is the one that should be accepted, either alter the local policy to accept the value in this proposal or notify the administrator of the remote security endpoint about the mismatch and ask the administrator to alter the remote configuration to propose the correct values. See the information about the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**User response**

Contact the system programmer.

**Module**

policy.cpp

**Procedure name**

None.

**EZD1116I**

IKE detected an NAPT in front of the remote security endpoint while initiating a new phase tunnel

**Explanation**

The Internet Key Exchange (IKE) daemon attempted to initiate a new Security Association (SA) with a remote security endpoint that is behind a NAT performing port translation (NAPT). The z/OS IKE daemon cannot initiate such a Security Association but can respond to negotiations with a remote security endpoint behind an NAPT.

A new SA of this configuration type is not supported because there might be problems with future negotiations and traffic flow. See the information about NAT traversal considerations in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.
Diagnosis Guide for more information. z/OS is providing NAT traversal support for a defined group of configurations where z/OS is running IKE. A description of the supported configurations is provided in configuration scenarios supported for NAT traversal in z/OS Communications Server: IP Configuration Guide.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:

**phase**

The phase (1 or 2) that the negotiation was in when the error occurred.

**System action**

The negotiation fails and all associated SAs are removed; IKE daemon processing continues.

**Operator response**

The z/OS IKE daemon can respond only to negotiations with a remote security endpoint behind an NAPT. Contact the administrator of the remote security endpoint to initiate the negotiation for this SA.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communication Server TCP/IP other application

**Module**

phase1.cpp

**Example**

None.

**EZD1117I** Initiation of a phase 2 negotiation with a remote security endpoint behind an NAPT is prohibited - the pending phase 2 request was deleted

**Explanation**

A request to initiate a phase 2 Security Association (SA) with a remote security endpoint behind a NAT performing port translation (NAPT) was deleted. A new SA of this configuration type is not supported because there might be problems with future negotiations and traffic flow. See the information about NAT traversal considerations in z/OS Communications Server: IP Diagnosis Guide for more information.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.
**System action**
The request for the phase 2 negotiation is deleted; IKE daemon processing continues.

**Operator response**
The z/OS IKE daemon can respond only to phase 2 negotiations with a remote security endpoint behind an NAPT. Request that the administrator of the remote security endpoint initiate the SA for this negotiation.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP other application

**Module**
phase1.cpp

**Example**
None.

---

**EZD1118I**  
Missing required keyword **keyword** for IKE configuration file parameter **pname** on line **linenum**

**Explanation**
The Internet Key Exchange (IKE) daemon configuration file contains a parameter that requires a keyword that was not found.

In the message text:

*keyword*
   - The keyword that was not found.

*pname*
   - The parameter that requires a keyword.

*linenum*
   - The line in the IKE daemon configuration file on which the error was found.

**System action**
If the error occurs during IKE startup, IKE daemon configuration file processing ends, and the IKE daemon ends. If the error occurs as a result of a MODIFY REFRESH command, IKE daemon configuration file processing ends, but the IKE daemon remains active. IKE configuration retains all values configured prior to the MODIFY REFRESH command.

**Operator response**
Contact the system programmer.
System programmer response
Specify the required keyword keyword and any value that is required for that keyword. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file and valid parameters for the keyword specified by the keyword value.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp

Routing code
10

Descriptor code
12

Example
EZD1118I Missing required keyword Identity for IKE configuration file parameter NetworkSecurityServer on line 14

EZD1119I Missing required parameter pname for statement sname in IKE configuration file fname on line linenum

Explanation
Line linenum of the IKE daemon configuration file fname contains a statement sname that requires a parameter pname that was not found.
In the message text:

pname
   The name of the missing parameter.

sname
   The name of the statement that requires the parameter.

fname
   The name of the IKE daemon configuration file.

linenum
   The line in the IKE daemon configuration file on which the error was found.

System action
If the error occurs during IKE startup, IKE daemon configuration file processing ends, and the IKE daemon ends. If the error occurs as a result of a MODIFY REFRESH command, IKE daemon configuration file processing ends, but the IKE daemon remains active. IKE configuration retains all values configured prior to the MODIFY REFRESH command.
Operator response
Contact the system programmer.

System programmer response
Specify the required parameter and any value that is required for that parameter. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file and valid parameters for the statement specified by the sname value.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp

Routing code
10

Descriptor code
12

Example
EZD1119I Missing required parameter ServiceType for statement NssStackConfig in IKE configuration file /etc/security/iked.conf on line 21

EZD1120I IKE configuration file contains more than one IkeConfig statement - values in last specified statement will be used

Explanation
The Internet Key Exchange (IKE) daemon configuration file contains more than one instance of the IkeConfig statement. Only one IkeConfig statement per IKE daemon configuration file is supported. The last instance encountered will be used.

System action
The values specified in the last IkeConfig statement are used. The IKE daemon continues.

Operator response
Contact the system programmer.

System programmer response
If this message was unexpected, see the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file and which statements and parameters are repeatable or non-repeatable.

EZD1xxx messages 749
User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp

Routing code
10

Descriptor code
12

Example
Not applicable.

EZD1121I  IKE configuration file contains more than one NssStackConfig statement for stack sname - values in last specified statement will be used

Explanation
The Internet Key Exchange (IKE) daemon configuration file contains more than one instance of the NssStackConfig statement for the same stack name. The NssStackConfig statement is a repeatable statement, but it is repeatable only for unique stack names.

In the message text:

sname
    The name of the stack for which multiple NssStackConfig statements were specified.

System action
The values of the last specified NssStackConfig statement are used for the specified stack; the IKE daemon continues.

Operator response
Contact the system programmer.

System programmer response
If this message was unexpected, see the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file and which statements and parameters are repeatable or non-repeatable.

User response
Not applicable.
**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
ike_config.cpp

**Routing code**
10

**Descriptor code**
12

**Example**
Not applicable.

**EZD1122I  Error initializing NMI - monitoring support is not available**

**Explanation**
An error during the startup of the IPSec network management interface (NMI) prevents the IKE daemon from accepting NMI requests.

Additional diagnostic messages that have the same message instance number will be issued to further identify the error. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

**System action**
The IKE daemon continues without NMI support.

**Operator response**
Not applicable.

**System programmer response**
If NMI support is needed, fix the underlying problem as indicated by other messages that have the same message instance number and restart the IKE daemon.

**User response**
Not applicable.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP: IKE daemon
Module
main.cpp

Routing code
10

Descriptor code
12

Example

Message instance 1: EZD0970I C/C++ runtime library function call pthread_create failure : 0001 - 12 | 01370010 | Not enough memory available
Message instance 1: EZD1122I Error initializing NMI - monitoring support is not available

**Ezd1123I**  NMI connection from user *username* closed - *num_conns* connections remain open

**Explanation**
A network management interface (NMI) connection has been closed. Some connections might remain open. NMI supports a maximum of 50 open connections.

In the message text:

*username*  
The user name from the closed NMI connection.

*num_conns*  
The total number of NMI connections from all users that remain open.

**System action**
The IKE daemon continues.

**Operator response**
Not applicable.

**System programmer response**
Not applicable.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

Module

Routing code
10
**Example**

<table>
<thead>
<tr>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1123I</td>
<td>NMI connection from user user33 closed - 10 connections remain open</td>
</tr>
</tbody>
</table>

**EZD1124I NMI connection received from user **USERNAME** - num_conns connections open**

**Explanation**

A new network management interface (NMI) connection was received and is ready to handle requests. NMI supports a maximum of 50 open connections.

In the message text:

- **USERNAME**
  The user name from the NMI connection that was received.

- **num_conns**
  The total number of NMI connections from all users that are open.

**System action**

The IKE daemon continues.

**Operator response**

Not applicable.

**System programmer response**

Not applicable.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

**Routing code**

10

**Descriptor code**

12

**Example**

<table>
<thead>
<tr>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1124I</td>
<td>NMI connection received from user <strong>USERNAME</strong> - 10 connections open</td>
</tr>
</tbody>
</table>

**EZD1125I SERVAUTH check for user **USERNAME** and profile **PROFILE** failed during an NMI request**

---

EVD1xxx messages 753
Explanation

A SERVAUTH check for the specified user failed when the IKE daemon was verifying access to the specified resource profile while processing a Network Management Interface (NMI) request.

In the message text:

**username**

The user name of the NMI client that issued the request

**profile**

The SERVAUTH profile being checked for the user for this NMI request.

System action

The NMI request is not processed and the connection remains open; the IKE daemon continues.

Operator response

Not applicable.

System programmer response

If the user should be allowed to access the specified resource, grant READ access to the specified profile in the SERVAUTH class. See the information about the IPSec network management interface access control in z/OS Communications Server: IP Configuration Guide for more information.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP: IKE daemon

Module

Routing code

10

Descriptor code

12

Example

```
EZD1125I SERVAUTH check for user user33 and profile EZB.NETMGMT.MVS052.MVS052.IKED.DISPLAY failed during an NMI request
```

EZD1126I Unable to send NMI message because of a memory shortage

Explanation

The Internet Key Exchange (IKE) daemon is unable to send a network management interface (NMI) message because there is not enough memory to build some of the required internal structures. Message EZD0963I is issued prior to this message to indicate the amount of memory that was being requested when the failure occurred.
**System action**
The NMI message is not sent and a termination message is sent over the connection on which the request was received. The IKE daemon continues.

**Operator response**
Free some memory and try the operation again. See the information about the diagnosing storage abends and storage growth in z/OS Communications Server: IP Diagnosis Guide for more information about storage problems.

**System programmer response**
Not applicable.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**

**Routing code**
10

**Descriptor code**
12

**Example**

```
EZD1126I Unable to send NMI message because of a memory shortage
```

```
EZD1127I NMI connection from user username closed - maximum number of NMI connections open
```

**Explanation**
The network management interface (NMI) closed a new connection from the specified user because the maximum number of connections are already open. NMI supports a maximum of 50 open connections.

In the message text:

*username*  
The user name from the NMI connection that was closed.

**System action**
An NMI termination message is sent over the new connection and the connection is closed; the IKE daemon continues.

**Operator response**
Not applicable.
System programmer response
Not applicable.

User response
Wait and try the request again when connections might be available.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module

Routing code
10

Descriptor code
12

Example
EZD1127I NMI connection from user33 closed - maximum number of NMI connections open

EZD1128I IKE STATUS FOR STACK stackname IS ACTIVE WITHOUT POLICY

Explanation
The Internet Key Exchange (IKE) daemon detected a status change for the specified stack. The IKE daemon can establish Security Associations only after message EZD1058I has been issued for a given stack. The IKE daemon can provide Network Monitoring Interface (NMI) support for a stack after message EZD1128I or message EZD1058I has been issued for a given stack. The IKE daemon might detect a stack status change for several reasons, such as encountering an NssStackConfig statement in the IkeConfig file. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the NssStackConfig statement.

In the message text:

stackname
The name of the stack for which a status change has been detected.

System action
The IKE daemon continues.

Operator response
None.

System programmer response
None.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
stackobj.cpp

Routing code
10

Descriptor code
12

Example
EZD1128I IKE STATUS FOR STACK TCPCS IS ACTIVE WITHOUT POLICY

EZD1129I RACF PassTicket generation failed

Explanation
The Internet Key Exchange (IKE) daemon could not generate the RACF PassTicket required to connect to the network security services (NSS) server.

System action
The connection request to the NSS server fails; the IKE daemon continues. Certificate and remote management services will not be available to the unauthorized client stack.

Operator response
Contact the system programmer.

System programmer response
See the information about IP security in z/OS Communications Server: IP Configuration Guide for information about PassTicket configuration for an NSS client.

For specific information about PassTickets, see the z/OS Security Server RACF Security Administrator's Guide.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon
**EZD1130I**  A *response_type* response message with correlator ID *corr_id* from the NSS server was discarded because a matching request was not found

**Explanation**
The Internet Key Exchange (IKE) daemon could not match the received response from the network security services (NSS) server with a corresponding request.

In the message text:

- **response_type**: The type of response received from the NSS server.
- **corr_id**: The 16-byte message correlator contained in the response message.

**System action**
The response message is discarded; the IKE daemon continues.

**Operator response**
No action needed.

**System programmer response**
One reason that this message might be issued is that the associated Security Association might have been deleted prior to receiving the response. If this was unexpected, See the information about diagnosing IP security problems in z/OS Communications Server: IP Diagnosis Guide for information about resolving connectivity problems.

**User response**
Contact the system programmer.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon
Routing code
10

Descriptor code
12

Example

EZD1130I A verify signature response message with correlator ID 0x0123456789ABCDEF from the Network Security Server was discarded because a matching request was not found

EZD1131I The IKE daemon will not connect to the NSS server for stackname because the stack is not configured with IPSECURITY support

Explanation
The iked.conf file has an NssStackConfig statement specified for this stack, but the stack is not configured with IPSECURITY support. The Internet Key Exchange (IKE) daemon connects to only the Network Security Services (NSS) server for stacks that are configured with IPSECURITY support.

In the message text:

stackname
The TCP/IP stack that is specified on the NssStack Config parameter in the iked.conf file.

System action
The IKE daemon does not connect to the NSS server for this stack. The IKE daemon continues.

Operator response
Notify the system programmer.

System programmer response
If this stack should connect to the NSS server, check the stack profile to ensure that IPSECURITY is configured correctly. If this stack should not connect to the NSS server, comment out the NssStackConfig parameter and its associated values from the iked.conf file to avoid seeing this message. See the information about IPCONFIG in z/OS Communications Server: IP Configuration Reference for more information about specifying IPSECURITY for a stack.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IP Sec

Module
stackobj.cpp
Routing code
12

Descriptor code
10

Example

EZD1131I The IKE daemon will not connect to the NSS server for TCPCS because the stack is not configured with IPSECURITY support

EZD1132I A request_type request for the NSScertificate services could not be sent for stack stack_name - return code = ret_code

Explanation
The Internet Key Exchange (IKE) daemon could not send a network certificate services request to the network security services (NSS) server.

In the message text:
request_type
The type of request.
stack_name
The name of the stack.
ret_code
Possible values are:
-1
The IKE daemon is not connected to the NSS server.
-2
The stack is not configured to use network security certificate services.
-3
The stack is not authorized to use network security certificate services.

System action
The request fails; the IKE daemon continues.

Operator response
No action needed.

System programmer response
If the return code indicates that the IKE daemon is not connected to the NSS server, See the information about the diagnosing network connectivity problems in z/OS Communications Server: IP Diagnosis Guide for information about resolving connectivity problems. If the return code indicates that the stack is not authorized to use network security certificate services, notify the system programmer of the NSS server to provide authorization to the stack for network security certificate services. See the information about IP security in z/OS Communications Server: IP Configuration Guide for information about network security certificate services.

User response
If the return code indicates that the stack is not configured to use network security certificate services, see the information about IP security in z/OS Communications Server: IP Configuration Guide for information about configuring network security services. Contact the system programmer for all other return codes.
Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
oakley_kep.cpp

Routing code
10

Descriptor code
12

Example
EZD1132I A verify signature request for network security certificate services could not be sent for stack TCPCS - return code -1

EZD1133I IKE STATUS FOR STACK stackname IS ACTIVE WITHOUT IPSECURITY SUPPORT

Explanation
The Internet Key Exchange (IKE) daemon detected that the specified stack is active but is not configured with IP security support. The IKE daemon can establish Security Associations for the specified stack only after that stack has specified the IPCONFIG IPSECURITY option in its configuration profile. See the information about the IPCONFIG statement in z/OS Communications Server: IP Configuration Reference for more information about the IPSECURITY parameter.

In the message text:

stackname
The name of the stack that is active without IP security support.

System action
The IKE daemon continues.

Operator response
None.

System programmer response
If you want IP security support, specify the IPCONFIG IPSECURITY option in the configuration profile of the stack. See the information about the IPCONFIG statement in z/OS Communications Server: IP Configuration Reference for more information about the IPSECURITY parameter.

User response
Not applicable.

Problem determination
Not applicable.
Source
z/OS Communications Server TCP/IP: IKE daemon

Module
stackobj.cpp

Routing code
10

Descriptor code
12

Example
EZD1133I IKE STATUS FOR STACK TCPCS IS ACTIVE WITHOUT IPSECURITY SUPPORT

EZD1135I NssStackConfig statements were specified, but no NSS server configuration was found in the IkeConfig statement

Explanation
One or more NssStackConfig statements were found, but the IkeConfig statement did not contain a NetworkSecurityServer parameter or NetworkSecurityServerBackup parameter.

System action
If the error occurs during IKE daemon startup, IKE daemon configuration file processing ends, and the IKE daemon ends. If the error occurs as a result of a MODIFY REFRESH command, IKE daemon configuration file processing ends, but the IKE daemon remains active using the configuration values that existed prior to the MODIFY REFRESH command.

Operator response
Contact the system programmer.

System programmer response
Either remove the NssStackConfig statements or add a NetworkSecurityServer parameter or NetworkSecurityServerBackup parameter to the IkeConfig statement. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file and these IkeConfig parameters.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp
Routing code
10

Descriptor code
12

Example
Not applicable.

**EZD1136I**  The IKE daemon is connected to the NSS server at *location* port *port* for stack *stackname*

**Explanation**
The Internet Key Exchange (IKE) daemon is connected to the network security services (NSS) server.

In the message text:

- **location**
  The configured hostname or IP address of the NSS server.

- **port**
  The port of the NSS server.

- **stackname**
  The name of the stack for which the IKE daemon has established a connection to the NSS server.

**System action**
The IKE daemon can use network security services for the specified stack; the IKE daemon continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
Not applicable.

Routing code
10

Descriptor code
12
The IKE daemon is connected to the network security server at 1.2.3.4 port 4519 for stack TCPCS

**EZD1137I** The IKE daemon is disconnected from the NSS server at location port port for stack stackname

**Explanation**
A connection from the Internet Key Exchange (IKE) daemon to the network security services (NSS) server has been disconnected.

In the message text:
- **location** The configured hostname or IP address of the NSS server.
- **port** The port of the NSS server.
- **stackname** The name of the stack for which the IKE daemon does not have an established connection to the NSS server.

**System action**
Network security services are not available for the specified stack; the IKE daemon continues.

**Operator response**
This message is issued only after message EZD1136I has been issued. The following are examples of when this message is issued:
- During IKE daemon shutdown.
- The NSS server is stopped while the IKE daemon is active.
- The IKE daemon configuration file is refreshed following deletion of the NetworkSecurityServer[Backup] statement or deletion of the NssStackConfig statement for the specified stack.

If this message is issued under any other circumstances, notify the system programmer.

**System programmer response**
See the information about the diagnosing IP security problems in z/OS Communications Server: IP Diagnosis Guide for information about resolving connectivity problems.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
Not applicable.

**Routing code**
10
The IKE daemon is disconnected from the network security server at 1.2.3.4 port 4519 for stack TCPCS.

EZD1138I  The IKE daemon is connecting to the NSS server at ipaddr port port for stack stackname

Explanation
The Internet Key Exchange (IKE) daemon is connecting to the network security services (NSS) server.

In the message text:

stackname
The stack for which the IKE deamon is establishing a connection to the NSS server.

ipaddr
The IP address of the NSS server.

port
The port of the NSS server.

System action
The IKE daemon continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
Not applicable.

Routing code
10

Descriptor code
12
Example

EZD1138I The IKE daemon is connecting to the network security server at 1.2.3.4 port 4503 for stack TCPCS

EZD1139I Request type requestcode with correlator ID corrid for stack stackname failed - return code returncode reason code reasoncode

Explanation

The Internet Key Exchange (IKE) daemon sent a network security services request to the network security services (NSS) server and the server sent a response indicating that the request failed.

In the message text:

requestcode
The NMsMType field of the response.

corrid
A unique identifier for the request.

stackname
The stack for which the request was made.

returncode
The NMsMRc field of the response

reasoncode
The NMsMRsn field of the response.

System action
The request fails; the IKE daemon continues.

Operator response
Contact the system programmer.

System programmer response
See the information about the error code descriptions in z/OS Communications Server: IP Diagnosis Guide to determine the appropriate response.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
Not applicable.

Routing code
10
EZD1139I Request type NSS_CreateSignatureReqToSrv for stack TCPCS with correlator ID 0x0123456789ABCDEF failed - return code EINVAL reason code NSSRsnNoMatchingCert

EZD1140I The NSS server identity is not valid

Explanation
The subject name or one of the subject alternate names in the certificate used by the network security services (NSS) server must match the Identity parameter on either the NetworkSecurityServer keyword or the NetworkSecurityServerBackup keyword of the IkeConfig statement.

The Internet Key Exchange (IKE) daemon will not use network security services and will not be available for remote management until the condition is resolved.

System action
The IKE daemon will periodically try to connect to the server specified by the NetworkSecurityServer keyword and the server specified by the NetworkSecurityServerBackup keyword as configured using the NssWaitLimit and NssWaitRetries keywords of the IkeConfig statement. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about these keywords.

Operator response
Contact the system programmer.

System programmer response
Check the NetworkSecurityServer keyword or the NetworkSecurityBackup keyword of the IkeConfig statement. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about these keywords. If these keywords are configured correctly, check the certificate used by the NSS server. The certificate subject name or one of the certificate subject alternate names must match the configured identity. See the information about the IPSec Certificate management in z/OS Communications Server: IP Configuration Guide for more information about managing NSS server certificates.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
Not applicable.

Routing code
10
The keyword identity ident does not match the subject name or any of the subject alternate names in the certificate used by the NSS server

**Explanation**
An identity mismatch was detected for the network security services (NSS) server.

In the message text:

**keyword**
Possible values are NetworkSecurityServer or NetworkSecurityServerBackup.

**ident**
The Identity parameter as configured on either the NetworkSecurityServer keyword or the NetworkSecurityBackup keyword of the IkeConfig statement.

**System action**
The IKE daemon continues.

**Operator response**
Check the log for message EZD1140I. If message EZD1140I is in the log, see that message for further information. If message EZD1140I is not in the log, no further action is necessary.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12
EZD1141I The NetworkSecurityServer identity 9.1.1.5 does not match the subject name or any of the subject alternate names in the certificate used by the network security server.
EZD1141I The NetworkSecurityServerBackup identity 9.1.2.6 does not match the subject name or any of the subject alternate names in the certificate used by the network security server.
EZD1140I The network security server identity is not valid.

**EZD1142I** Name resolution failed for *keyword* hostname *hostname*

**Explanation**
Name resolution failed for the specified host name on either the NetworkSecurityServer keyword or the NetworkSecurityServerBackup keyword.

In the message text:

*keyword*
Possible values are NetworkSecurityServer or NetworkSecurityBackup.

*hostname*
The name that was configured on the host parameter on either the NetworkSecurityServer keyword or the NetworkSecurityServerBackup keyword.

**System action**
Name resolution fails; the IKE daemon continues. The IKE daemon will not connect to the network security services (NSS) server that corresponds to the specified keyword.

**Operator response**
Contact the system programmer.

**System programmer response**
Verify that the host parameter on the NetworkSecurityServer keyword or the NetworkSecurityServerBackup keyword is the correct name. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about these keywords. See the information about diagnosing resolver problems information in z/OS Communications Server: IP Diagnosis Guide for information about resolving name resolution problems. Restart the IKE daemon or issue the MODIFY IKED,REFRESH command to attempt the name resolution again. See z/OS Communications Server: IP System Administrator's Commands for more information about the MODIFY command.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
Not applicable.

**Routing code**
10
Descriptor code
12

Example
Not applicable.

**EZD1143A**  The IKE daemon cannot locate the NSS server

**Explanation**
The Internet Key Exchange (IKE) daemon could not resolve the NetworkSecurityServer host parameter or the NetworkSecurityServerBackup host parameter to an IP address.

**System action**
The IKE daemon will not connect to the network security services (NSS) server until the problem is resolved; the IKE daemon continues.

**Operator response**
Contact the system programmer.

**System programmer response**
This message is accompanied by message EZD1142I. See the system programmer response for message EZD1142I to resolve this problem.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
Not applicable.

**Routing code**
1

**Descriptor code**
2

**Example**
Not applicable.

**EZD1144I**  The NSS certificate service is available for stack stackname
Explanation
The network security services (NSS) server authorized the IKE daemon to use the certificate service for the stack.

In the message text:

stackname
The name of the stack for which the network security certificate service is available.

System action
The IKE daemon will support digital signature modes of authentication for the stack; the IKE daemon continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
stackobj.cpp

Routing code
10

Descriptor code
12

Example
EZD1144I The network security certificate service is available for stack TCPIP

EZD1145I The NSS certificate service is not available for stack stackname

Explanation
The network security services (NSS) server did not authorize the IKE daemon to use the certificate service for the stack.

In the message text:

stackname
The name of the stack for which the certificate service is not available.
**System action**
The IKE daemon will not support digital signature modes of authentication for the stack; the IKE daemon continues.

**Operator response**
Contact the system programmer.

**System programmer response**
See information about the network security services (NSS) server authorization considerations in z/OS Communications Server: IP Configuration Guide for information about authorizing NSS clients.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
stackobj.cpp

**Routing code**
10

**Descriptor code**
12

**Example**

```
EZD1145I The network security certificate service is not available for stack TCPIP
```

**EZD1146I**  The NSS remote management service is available for stack *stackname*

**Explanation**
The network security services (NSS) server authorized the IKE daemon to use the remote management service for the stack.

In the message text:

*stackname*

  The name of the stack for which the network security remote management service is available.

**System action**
The IKE daemon will support network security remote management for the stack; the IKE daemon continues.

**Operator response**
None.
System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
stackobj.cpp

Routing code
10

Descriptor code
12

Example
EZD1146I The NSS remote management service is available for stack TCPIP

EZD1147I The NSS remote management service is not available stack stackname

Explanation
The network security services (NSS) server did not authorize the IKE daemon to use the remote management service for the stack.

In the message text:
stackname
The name of the stack for which the network security remote management service is not available.

System action
The IKE daemon will not support network security remote management for the stack; the IKE daemon continues.

Operator response
Contact the system programmer.

System programmer response
See the information about the network security server authorization considerations in z/OS Communications Server: IP Configuration Reference for information about authorizing NSS clients.

User response
Not applicable.
Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
stackobj.cpp

Routing code
10

Descriptor code
12

Example
EZD1147I  The network security remote management service is not available for stack TCPIP

EZD1148A  THE IKE DAEMON CANNOT CONNECT TO THE NSS SERVER FOR stackname BECAUSE THE STACK DOES NOT SUPPORT IPV6

Explanation
The network security server is reachable using only an IPv6 address but the stack does not support IPv6.

In the message text:
stackname
   The name of the stack for which the IKE daemon cannot connect to the network security server.

System action
The IKE daemon will not connect to the network security services (NSS) server for the specified stack until the problem is resolved; the IKE daemon continues.

Operator response
Contact the system programmer.

System programmer response
Either reconfigure the stack to support IPv6 or reconfigure the NSS server location using an IPv4 address. For information about reconfiguring the stack to support IPv6, see the information about the Enabling IPv6 support in z/OS Communications Server: IPv6 Network and Application Design Guide.

For information about configuring the network security server location, use the host parameter of the NetworkSecurityServer and NetworkSecurityServerBackup keywords. See the information about the IKE daemon information in z/OS Communications Server: IP Configuration Reference for more information about these keywords. If the host parameter on either the NetworkSecurityServer or the NetworkSecurityServerBackup keywords indicates a hostname, reconfigure the domain name server so that at least one IPv4 address is resolved for the hostname. See the information about the diagnosing resolver problems in z/OS Communications Server: IP Diagnosis Guide for information about resolving name resolution problems. Restart the IKE daemon or issue the MODIFY IKED,REFRESH command to attempt the name resolution again. See the information about the MODIFY command in z/OS Communications Server: IP System Administrator's Commands for more information.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
Not applicable.

Routing code
1

Descriptor code
2

Example
None.

EZD1149I  The IKE daemon connection to the NSS server at ipaddr port port for stack stackname is not secure

Explanation
The Internet Key Exchange (IKE) daemon connection to the network security server is not secure. The connection to the network security server must be secured using Application Transparent Transport Layer Security (AT-TLS).

In the message text:

ipaddr  The IP address of the network security server.

port  The port of the network security server

stackname  The name of the stack for which the IKE daemon connection to the network security server is not secure.

System action
Network security services are not available for the specified stack; the IKE daemon continues. The IKE daemon will attempt to connect to the NetworkSecurityServer and the NetworkSecurityServerBackup in a round-robin fashion until a secure connection is established. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about these keywords.

Operator response
Notify the system programmer.
System programmer response

See the information about configuring the IKE daemon information in z/OS Communications Server: IP Configuration Guide for information about defining AT-TLS policy to protect communication with a network security server.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP: IPSec

Module

Not applicable.

Routing code

10

Descriptor code

12

Example

EZD1149I The IKE daemon connection to the network security server at 1.9.0.1 port 4159 for stack TCPCS is not secure

EZD1150I THE IKE DAEMON FAILED TO CONNECT retries TIMES TO THE server_type NSS SERVER AT host PORT port FOR STACK stackname

Explanation

The Internet Key Exchange (IKE) daemon was unable to establish a connection to the Network Security Services (NSS) server for the stack, host, and port specified for the number of retries specified on the NssWaitRetries parameter of the iked.conf file.

In the message text:

retries
  The number of retries attempted.

server_type
  The type of NSS server for which a connection was not established. Possible values are:
  PRIMARY
    Primary NSS server
  BACKUP
    Backup NSS server

host
  The configured hostname or IP address for which the connection to the NSS server was not established.

port
  The port for which the connection to the NSS server was not established.
**stackname**
The name of the TCP/IP stack for which the connection to the NSS server was not established.

**System action**
IKED continues trying to connect to the NSS server by doing one of the following:
- Switching servers, if both a primary and a backup NSS server have been specified in the iked.conf file.
- Trying to connect to the same server, if only a primary or a backup NSS server has been specified in the iked.conf file.

The IKE daemon continues.

**Operator response**
If connectivity cannot be established, notify the system programmer.

**System programmer response**
Ensure that the NetworkSecurityServer and NetworkSecurityServerBackup parameters in the iked.conf file have been configured correctly. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about these keywords. If these keywords are configured correctly, ensure the TCP/IP configuration is specified correctly.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IPSec

**Module**
IkeInetConnectManager.cpp

**Routing code**
10

**Descriptor code**
12

**Example**
```
EZD1150I THE IKE DAEMON FAILED TO CONNECT 4 TIMES TO THE PRIMARY NSS SERVER AT 2.5.2.6 PORT 4159 FOR STACK TCPCS4
```

EZD1151I KeyExchangeAction actionname prevents the creation of a dynamic tunnel with source data endpoint specification source_ip and destination data endpoint specification dest_ip

**Explanation**
Dynamic tunnel activation is denied as a result of a configured source or destination data IP address constraint. See the information about the KeyExchangeAction statement in z/OS Communications Server: IP Configuration Reference for an explanation of data address constraints.
In the message text:

**actionname**  
The name of the KeyExchangeAction statement configured with a source or destination IP address constraint.

**source_ip**  
The source IP address of the dynamic tunnel.

**dest_ip**  
The destination IP address of the dynamic tunnel.

**System action**  
The dynamic tunnel activation fails; IKE daemon processing continues.

**Operator response**  
Contact the system programmer.

**System programmer response**  
If the tunnel activation should be permitted, then do one of the following to correct the configuration.

- When IPSec policy is configured with the IBM Configuration Assistant for z/OS Communications Server, add a connectivity rule with a local data endpoint that matches the `source_ip` value and a remote data endpoint that matches the `dest_ip` value at the top of the rule list.
- When IPSec policy is configured without the IBM Configuration Assistant for z/OS Communications Server, update the KeyExchangeAction ConstrainSource or ConstrainDest configuration to include the `source_ip` value and the `dest_ip` value. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

If the tunnel activation should not be permitted, then determine whether the tunnel was activated locally or remotely. If the `source_ip` value matches the IP address value in the Local IPSec Client ID information from the Security Association (SA) Context Information that was output with the message, then the tunnel was activated locally. Otherwise, the tunnel was activated remotely. If the tunnel was activated locally but should not be permitted, then correct the local IpFilterPolicy statement to block the activation. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy. If the tunnel was activated remotely but should not be permitted, then contact the owner of the remote system to request that the activation be blocked on that system.

**User response**  
Not applicable.

**Problem determination**  
Not applicable.

**Source**  
z/OS Communications Server TCP/IP: IKE daemon

**Module**  
policymgr.cpp

**Routing code**  
Not applicable for syslog message.
Descriptor code

Not applicable for syslog message.

Automation

This message goes to the syslog.

Example

<table>
<thead>
<tr>
<th>Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: *** SA Context Information ***</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: Phase 2 SAID : 0</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: Assoc P1 ID : 2</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: Stackname : TCPCS1</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: Local IPSec Client ID info : Ipv4 1.1.0.1 Port: Any</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: Remote IPSec Client ID info : IPV4 Subnet 0.0.0.0/0 Port: Any</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: Local IPSec IP info : 1.1.0.1</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: Remote IPSec IP info : 1.2.0.1</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: LocalDynVpnRuleName : udpvpn1</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: AH SPIs in/out : 0 / 0</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: ESP SPIs in/out : 0 / 0</td>
</tr>
<tr>
<td>Jun 19 22:34:08</td>
<td>MVS073 IKE: Message instance 3: EZD1151I KeyExchangeAction ConstrainedAction1 prevents the creation of a dynamic tunnel with source data endpoint specification 1.1.0.1 and destination data endpoint specification 0.0.0.0/0</td>
</tr>
</tbody>
</table>

EZD1152I The IKE daemon is resolving server hostname name

Explanation

The Internet Key Exchange (IKE) daemon is resolving a server hostname. The IKE daemon must resolve the hostname before it can connect to the Network Security Services (NSS) server. No information is exchanged with the NSS server while name resolution is in progress. Message EZD1153I is issued when the name resolution completes.

In the message text:

**server**

The server value can be either NetworkSecurityServer or NetworkSecurityServerBackup.

**name**

The host name being resolved.

System action

The IKE daemon continues.

Operator response

If the EZD1152I message is not followed by an EZD1153I message it might indicate a problem with the resolver. Contact the system programmer.

System programmer response

If the EZD1152I message is not followed by an EZD1153I message then ensure that the resolver is operating correctly. See the information about diagnosing resolver problems information in z/OS Communications Server: IP Diagnosis Guide for more information.

User response

Not applicable.
Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IPSec

Module
Not applicable.

Routing code
10

Descriptor code
12

Example

EZD1152I The IKE daemon is resolving the NetworkSecurityServer hostname mvs073
EZD1153I The IKE daemon resolved the NetworkSecurityServer hostname mvs073 to 3 addresses

Explanation
The Internet Key Exchange (IKE) daemon resolved the Network Security Services (NSS) server hostname to zero or more addresses. The IKE daemon will not connect to the NSS server unless the hostname resolves to one or more addresses.

In the message text:

server
The server value can be either NetworkSecurityServer or NetworkSecurityServerBackup.

name
The host name that was resolved.

count
The number of addresses to which the hostname resolved.

System action
The IKE daemon continues.

Operator response
If the number of addresses is zero, contact the system programmer.

System programmer response
If the number of addresses is zero, correct the configuration so that the server hostname resolves to one or more addresses. See the information about resolvers in z/OS Communications Server: IP Configuration Guide for more information.

User response
Not applicable.
Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IPSec

Module
Not applicable.

Routing code
10

Descriptor code
12

Example

```plaintext
EZD1152I The IKE daemon is resolving the NetworkSecurityServer hostname mvs073
EZD1153I The IKE daemon resolved the NetworkSecurityServer hostname mvs073 to 3 addresses
```

```
EZD1154I The remote data endpoint rip with identity rid does not match the remote security endpoint rsip using KeyExchangeAction actionname
```

Explanation
When the FilterByIdentity Yes parameter is configured on a KeyExchangeAction statement or a mobile user connectivity rule is configured using the IBM Configuration Assistant for z/OS Communications Server, the peer is restricted to negotiating data protection only for its security endpoint address. Remote identity support is intended for mobile users, who are not permitted to function as a security gateway. However, the peer’s data endpoint and security endpoint do not match, which violates the local restriction for mobile user negotiations. It is likely that the peer is not a mobile user; initiation is denied.

In the message text:

- **rip**
  The data endpoint IP address of the peer.

- **rid**
  The identity of the peer.

- **rsip**
  The security endpoint IP address of the peer.

- **actionname**
  The name of the KeyExchangeAction statement configured with FilterByIdentity Yes.

System action
The dynamic tunnel activation fails; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
If the peer at the security endpoint IP address should be permitted to function as a security gateway, then correct the local configuration. It is also possible that the peer is a mobile user but is configured in violation of the local restriction for mobile user negotiations. Contact the owner of the peer to determine whether the peer is
a mobile user. If so, inform the owner that the peer data endpoint IP address must be configured to match the peer security endpoint IP address when initiating a dynamic tunnel to the z/OS IKE daemon.

When IPSec policy is configured without the IBM Configuration Assistant for z/OS Communications Server, either update the KeyExchange action identified in the message to specify the FilterByIdentity N parameter or configure a new dynamic tunnel that is specific to the initiating peer. See the information about configuring the branch office model in z/OS Communications Server: IP Configuration Guide.

When IPSec policy is configured with the IBM Configuration Assistant for z/OS Communications Server, either edit the mobile user rule and change it to a typical rule, or create a new typical connectivity rule that is specific to the initiating peer and move the new rule above the mobile user rule. See the online helps in the GUI for additional information.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

doi.cpp

**Routing code**

Not applicable for syslog message.

**Descriptor code**

Not applicable for syslog message.

**Automation**

This message goes to the syslog.

**Example**

```
Jun 19 22:46:55 MVS073 IKE: Message instance 3: Phase 2 SAID : 0          Assoc P1 ID : 1
Jun 19 22:46:55 MVS073 IKE: Message instance 3: Local IPSec Client ID info  : IPV4 Subnet 0.0.0.0/0
Jun 19 22:46:55 MVS073 IKE: Message instance 3: Remote IPSec Client ID info : IPV4 Subnet 172.16.0.0/12 Port: Any
Jun 19 22:46:55 MVS073 IKE: Message instance 3: Remote IPSec Client ID info : IPV4 Subnet 172.16.0.0/12 Port: Any
Jun 19 22:46:55 MVS073 IKE: Message instance 3: AH SPIs in/out : 0 / 0
Jun 19 22:46:55 MVS073 IKE: Message instance 3: ESP SPIs in/out : 0 / 0
Jun 19 22:46:55 MVS073 IKE: Message instance 3: EZD1154I The remote data endpoint 172.16.0.0/12 with identity 1.1.0.1 does not match the remote security endpoint 1.1.0.1 using KeyExchangeAction strong-KE
```

**EZD1155I**  _t_name transform t_num in proposal p_num does not include an integrity algorithm**
**Explanation**

The Internet Key Exchange (IKE) daemon received a proposal that contains a transform with no encryption algorithm and no integrity algorithm during a dynamic tunnel negotiation. Such a proposal is not permitted and is an auditable event. If no acceptable proposal is received then EZD1022I will also be issued.

In the message text:

- **t_name**
  The transform name.

- **t_num**
  The transform number. There might be multiple transforms in a proposal.

- **p_num**
  The proposal number. There might be multiple proposals in a Security Association (SA) establishment message.

**System action**

If the IKE daemon does not accept any of the proposals, the negotiation fails; IKE daemon processing continues.

**Operator response**

Contact the system programmer.

**System programmer response**

Use the remote IPSec IP information in the SA context information to identify the source of the invalid proposal. Contact the owner of the invalid proposal and request that the configuration be corrected.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

ipsec_match.cpp

**Routing code**

Not applicable for syslog message.

**Descriptor code**

Not applicable for syslog message.

**Automation**

This message goes to the syslog.

**Example**

```
Jun 22 21:24:13 MVS073 IKE: Message instance 3: Phase 2 SAID : 4     Assoc P1 ID : 1
```
Extraneous text ignored on line *linenum* after *keyword* value

Explanation
The Internet Key Exchange (IKE) daemon encountered extraneous text in a configuration file. The extraneous text appeared after a keyword and value specification.

In the message text:

**linenum**
The line of the configuration file where the extraneous text was found

**keyword**
The configuration keyword after which the extraneous text was found

**value**
The configuration value after which the extraneous text was found

System action
The configuration keyword is processed, and the extraneous text is ignored. IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Remove the extraneous text from the indicated line in the IKE configuration file. See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for more information about the IKE daemon configuration file syntax.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IPSec

**Module**
ike_config.cpp
Routing code
Not applicable for syslog message.

Descriptor code
Not applicable for syslog message.

Automation
This message goes to the syslog.

Example
EZD1156I Extraneous text ignored on line 12 after IkeSyslogLevel 1
EZD1157I IKE message received from remote_ip port remote_port to local_ip port local_port with length message_length is too short to contain the IKE header

Explanation
The Internet Key Exchange (IKE) daemon received a message that is too short to contain the message header, which is 28 bytes long.

In the message text:
- **remote_ip**
  The remote security endpoint IP specification.
- **remote_port**
  The remote port of the IKE daemon peer.
- **local_ip**
  The local security endpoint IP specification.
- **local_port**
  The local port of the IKE daemon.
- **message_length**
  The length of the IKE message, in bytes.

System action
The SA negotiation fails; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that a protocol error has occurred.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon
Module
simple_net.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1157I IKE message received from 1.2.3.4 port 500 to 5.6.7.8 port 500 with length 27 is too short to contain the IKE header

EZD1158I DISPLAY IKE type:

Explanation
The Internet Key Exchange (IKE) daemon received the MODIFY DISPLAY subcommand.

In the message text:

*type*  
The type of IKE data that is displayed.

System action
The IKE daemon continues processing the MODIFY DISPLAY command.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
mdfysrvr.cpp

Routing code
2
**Example**

```
EZD1158I DISPLAY IKE CONFIGURATION:

EZD1159I Remote security endpoint id_type identity length id_length is longer than the maximum of id_max
```

**Explanation**

An IKE Security Association (SA) negotiation failed because the remote security endpoint identity received that was from the IKE peer was longer than the maximum allowed.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:

- **id_type**
  The name of the remote security endpoint identity type

- **id_length**
  The length of the remote security endpoint identity received from the IKE peer.

- **id_max**
  The maximum acceptable length for a remote security endpoint identity.

**System action**

The SA negotiation fails; IKE daemon processing continues.

**Operator response**

Contact the system programmer.

**System programmer response**

Notify the administrator of the remote security endpoint and ask the administrator to configure the remote security endpoint with a smaller identity or a different identity type.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

fw_convert.cpp
Routing code
*

Descriptor code
*

Automation
This message is output to syslog.

Example
Message instance 2: *** SA Context Information ***
Message instance 2: Phase 1 tunnel ID : K0 Generation : 0
Message instance 2: Stackname : TCPCS
Message instance 2: Local IKE ID info : ID_FQDN mvsas.tcp.raleigh.ibm.com
Message instance 2: Local IKE IP : 9.42.105.138 port 500
Message instance 2: Remote IKE IP : 9.42.141.122 port 500
Message instance 2: KeyExchangeRuleName : RSA
Message instance 2: Icookie/Rcookie : x82F0255E3A5993B2 / x58DFAFF2C1AD3F19
Message instance 2: IKE Version : 2
Message instance 2: EZD1159I Remote security endpoint ID_DER_ASN1_DN identity length 2285 is longer than the maximum of 1024

EZD1160I  Policy mismatch: IpDynVpnAction statement_name requires parameter parameter_name with value policy_value but the value selected by the IKE peer is peer_value

Explanation
The Internet Key Exchange (IKE) daemon was unable to accept a value selected by the IKE peer because the value is not allowed by the local policy. The Security Association negotiation will fail. Message EZD1022I will be issued to syslog and will indicate the failure.

In the message text:

statement_name

• In the policy agent configuration file, the statement_name value is the name specified on the applicable IpDynVpnAction statement.
• If the policy agent is configured with the IBM Configuration Assistant for z/OS Communications Server, the statement_name value corresponds to the name of the security level in the GUI. The value also contains a numeric suffix appended to the security level name to guarantee uniqueness.

parameter_name
The IpDynVpnAction parameter that encountered a mismatch. See the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about the parameter specified.

policy_value
The IpDynVpnAction parameter value that does not match the value selected by the IKE peer.

peer_value
The value selected by the IKE peer that does not match the policy_value value.

System action
The negotiation fails; the IKE daemon continues.

Operator response
Notify the system programmer.
System programmer response

Ensure that the IpDynVpnAction statement is configured correctly. Alter either the local policy to accept the value specified by peer_value in this statement or notify the administrator of the remote security endpoint about the mismatch and ask the administrator to alter the remote configuration to propose the policy_value value required by the local policy. See the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP: IKE daemon

Module

IKEv2TSRequest.cpp

Routing code

2

Descriptor code

5

Automation

Not applicable.

Example

EZD1160I Policy mismatch: IpDynVpnAction p2_action requires parameter HowToEncapIKEv2 with value Transport but the value selected by the IKE peer is Tunnel

EZD1161I DCAS CONFIGURATION SERVERTYPE IS UNDEFINED

Explanation

The combination of SERVERTYPE keyword definitions in the DCAS configuration file caused the SERVERTYPE keyword to be undefined to DCAS.

System action

DCAS ends.

Operator response

Contact the system programmer.

System programmer response

See z/OS Communications Server: IP Configuration Reference for a list of the SERVERTYPE keyword values in the DCAS configuration file.
Module
dcasconf.c

Procedure name
process_config_keyword()

EZD1162I  DCAS CONFIGURATION SERVERTYPE value IS NOT SUPPORTED

Explanation
This message is issued when DCAS is processing the DCAS configuration file. value is the SERVERTYPE keyword that is not supported.

System action
DCAS ends.

Operator response
Contact the system programmer.

System programmer response
See z/OS Communications Server: IP Configuration Reference for a list of the SERVERTYPE keyword values in the DCAS configuration file.

Module
dcasconf.c

Procedure name
process_config_keyword()

EZD1163I  ip_version DYNAMIC XCF INTERFACE xcfinterface_name WAS CREATED BUT THE SPECIFIED SOURCEVIPINTERFACE interface_name WILL NOT BE USED

Explanation
The interface name specified by the SOURCEVIPINTERFACE parameter on the IPCONFIG DYNAMICXCF statement or the IPCONFIG6 DYNAMICXCF statement was not valid. The interface name must be an active static VIPA link or interface name of the appropriate IP version.

In the message text:

ip_version
  Either IPV4 or IPV6

xcfinterface_name
  The Dynamic XCF interface name that was created

interface_name
  The name that was specified on the SOURCEVIPINTERFACE parameter for IPCONFIG DYNAMICXCF or IPCONFIG6 DYNAMICXCF

System action
The dynamic XCF interface is created. TCP/IP continues.
Operator response
Contact the system programmer.

System programmer response
Create a static VIPA with an interface name that matches the one specified on the IPCONFIG DYNAMICXCF or IPCONFIG6 DYNAMICXCF SOURCEVIPAINTERFACE statement and restart TCP/IP.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Configuration & Initialization

Module
EZBXFDYN

Routing code
2, 8

Descriptor code
12

Automation
Not applicable.

Example
EZD1163I IPV4 DYNAMIC XCF INTERFACE EZAXCFAA WAS CREATED BUT THE SPECIFIED SOURCEVIPAINTERFACE VIPA1 WILL NOT BE USED

EZD1164I IPV6 TCPSTACKSOURCEVIPA INTERFACE interface_name WAS NOT USED BY tcp_jobname

Explanation
An outbound connection request was processed but the interface configured with TCPSTACKSOURCEVIPA could not be used to determine the source IPv6 address because the interface was not an active static VIPA interface or an active dynamic VIPA interface.

interface_name is the name specified on the TCPSTACKSOURCEVIPA parameter of the IPCONFIG6 statement.

tcp_jobname is the name of the job associated with the procedure that was used to start TCP/IP.

System action
TCP/IP continues. To avoid flooding the system console, this informational message will not be issued again for at least five minutes.

Operator response
Contact the system programmer.
System programmer response
Change the TCPSTACKSOURCEVIPA interface name to be an active static or dynamic VIPA interface.

Module
EZBX6UTL

Procedure name
EZBX6SSV

EZD1165I  DVIPA INTERFACE interface_name IS ALREADY DEFINED WITH ip_addr

Explanation
The TCPIP detected that the interface interface_name was already defined as a DVIPA interface with a conflicting definition.

For a VIPADEFINE/VIPABACKUP configuration statement, the interface was already defined either by a previous VIPADEFINE/VIPABACKUP with a different IP address, or by a SIOCSVIPA IOCTL or BIND to an IP address in a configured VIPARANGE.

interface_name is the interface name specified on the VIPADEFINE/VIPABACKUP statement.
ip_addr is the IP address specified on the previous VIPADEFINE/VIPABACKUP statement.

System action
Processing continues. The VIPABACKUP or VIPADEFINE statement is rejected.

Operator response
Contact the system programmer.

System programmer response
Correct the VIPABACKUP or VIPADEFINE statement with a different IP address or with a different interface name.

Module
EZBX6DVI

Procedure name
PreValidateVBKUP6, PreValidateVDEF6

EZD1166E  tcpstackname DELAYING SYSPLEX PROFILE PROCESSING - application IS NOT ACTIVE

Explanation
The TCP/IP stack delayed joining a sysplex group and delayed processing sysplex definitions in the profile (VIPADYNAMIC and IPCONFIG/IPCONFIG6 DYNAMICXCF statements).

tcpstackname is the name of the TCP/IP stack.

application is the name of the application that is not active and is either OMPROUTE or VTAM.

See the information about sysplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide for more information.
System action
TCP/IP continues.

Operator response
If *application* is VTAM, start VTAM.
If the *application* value is OMPROUTE and it is not active, start OMPROUTE; If OMPROUTE is active, contact the system programmer. This message is issued for OMPROUTE only if GLOBALCONFIG SYSPLEXMONITOR DELAYJOIN was specified in a profile.
If you want the TCP/IP stack to immediately join a sysplex group rather than waiting for OMPROUTE to activate, issue the VARY TCPIP,,OBEYFILE command with GLOBALCONFIG SYSPLEXMONITOR NODELAYJOIN specified. See z/OS Communications Server: IP Configuration Reference for information about the DELAYJOIN keyword.
When OMPROUTE activation is complete or a VARY TCPIP,,OBEYFILE command with GLOBALCONFIG SYSPLEXMONITOR NODELAYJOIN has been specified, TCP/IP will join a sysplex group and finish processing the sysplex definitions.

System programmer response
If VTAM or OMPROUTE cannot be started, contact IBM software support services with the system log.

Module
EZBXFPDM

Procedure name
EZBXFPDM

EZD1167I DVIPA INTERFACE interface_name IS CONFIGURED FROM A VIPARANGE

Explanation
The interface name specified on a VIPADELETE was defined on a VIPARANGE statement. The interface cannot be deleted by using a VIPADELETE statement.
*interface_name* is the interface name specified on the VIPADELETE statement.

System action
Processing continues. The VIPADELETE statement is rejected.

Operator response
Contact the system programmer.

System programmer response
Change the VIPADELETE to specify a valid interface name, or use a VIPARANGE DELETE (instead of a VIPADELETE) to delete the configured VIPARANGE statement.

Module
EZBX6DVI

Procedure name
ValidateVDEL6

EZD1168I VIPADISTRIBUTE WITH THE PORT KEYWORD REJECTED FOR INTERFACE interface_name

EZD1xxxx messages 793
Explanation
The PORT keyword was specified on a VIPADISTRIBUTE statement for a Dynamic VIPA (DVIPA) that already had a VIPADISTRIBUTE statement specified without a PORT keyword, indicating dynamic ports.

*interface_name* is the interface name specified on the rejected VIPADISTRIBUTE statement with the PORT keyword.

System action
Processing continues. The VIPADISTRIBUTE statement is rejected.

Operator response
Contact the system programmer.

System programmer response
To disable dynamic ports, delete all previous VIPADISTRIBUTE statements for this DVIPA. Then reissue the VIPADISTRIBUTE with the PORT keyword.

Module
EZBX6DVI

Procedure name
ValidateVDIST6

EZD1169I  VIPADISTRIBUTE WITHOUT THE PORT KEYWORD REJECTED FOR INTERFACE
*interface_name*

Explanation
The PORT keyword was not specified on a VIPADISTRIBUTE statement indicating dynamic ports for a Dynamic VIPA (DVIPA) that already had a VIPADISTRIBUTE statement specified with a PORT keyword.

*interface_name* is the interface name specified on the rejected VIPADISTRIBUTE statement without the PORT keyword.

System action
Processing continues. The VIPADISTRIBUTE statement is rejected.

Operator response
Contact the system programmer.

System programmer response
To enable dynamic ports, delete all previous VIPADISTRIBUTE statements for this DVIPA. Then reissue the VIPADISTRIBUTE without the PORT keyword.

Module
EZBX6DVI

Procedure name
ValidateVDIST6

EZD1170E  tcpstackname WAS NOT ABLE TO GET TCP/IP *storagetype* STORAGE
Explanation
TCP/IP was not able to satisfy a request for storage.

`tcpstackname` is the name of the TCP/IP stack.

`storagetype` is the type of storage that was unavailable. The value for `storagetype` is either private or ECSA (Extended Common Storage Area)

System action
TCP/IP continues.

- If the GLOBALCONFIG SYSPLEXMONITOR RECOVERY option is active and this stack is not the only member of the TCP/IP sysplex group, the following RECOVERY actions will occur:
  - This stack will leave the TCP/IP sysplex group.
  - This stack will no longer participate in sysplex distribution (as a distributor or target) or act as an owner or a backup for DVIPAs. All DVIPAs defined on this stack will be deactivated; however, the DVIPA definitions will be saved.
  - When the stack leaves the TCP/IP sysplex group, this operator message will be deleted.
- If the GLOBALCONFIG SYSPLEXMONITOR NORECOVERY option is active, no action will be taken.

See the information about sysplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide for more information. See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for the definitions of the SYSPLEXMONITOR parameters.

Operator response
Save the TCP/IP profile and system log. If a dump was not created, then take a dump of the TCP/IP address space and dataspaces.

If NORECOVERY is active, no further actions are needed.

If RECOVERY is active, then even if the GLOBALCONFIG SYSPLEXMONITOR AUTOREJOIN option is active, the stack will not automatically rejoin the TCP/IP sysplex group, due to the severity of the problem encountered. Message EZZ9676E will be displayed if the TCP/IP stack successfully deactivates all DVIPAs and leaves the TCP/IP sysplex group. After this message is displayed, issuing the VARY TCPIP,,SYSPLEX,JOINGROUP command will cause the DVIPA definitions to be processed, and the stack to rejoin the TCP/IP sysplex group.

System programmer response
Use the display TCP/IP storage command `D TCPIP,,STOR` to determine the current status of TCP/IP private and ECSA storage. See z/OS Communications Server: IP System Administrator’s Commands for more information about the `D TCPIP,,STOR` command.

For ECSA storage exhaustion, determine which jobs or address spaces are using an excessive amount of storage. To determine the users of ECSA storage, enable common storage tracking. See the z/OS MVS Initialization and Tuning Guide for information about requesting common storage tracking. Use the VERBEXIT VSDATA OWNCOMM SUMMARY command to determine how much storage is used by each job. See the z/OS MVS Diagnosis: Tools and Service Aids for information about the IPCS VERBEXIT VSDATA command.

If the storage problem cannot be corrected, contact IBM software support services with all supporting documentation.

If the storage problem can be corrected:

- If RECOVERY is active, then issue the `VARY TCPIP,,SYSPLEX,JOINGROUP` command to cause the DVIPA definitions to be processed, and the stack to rejoin the TCP/IP sysplex group.

Module
EZBXFPDC
**EZD1171I** THERE IS NO ROUTE AVAILABLE FOR VIPAROUTE `dxcf_address target_ipaddress`

**Explanation**
The target stack identified by `dxcf_address` is active, and `target_ipaddress` is defined at that target stack, however no route is available to `target_ipaddress`. As a result, the local stack cannot forward any DVIPA packets to the target stack.

`dxcf_address` is the dynamic XCF address of a target stack as specified on the VIPAROUTE statement.

`target_ipaddress` is the IP address in the HOME list of the target stack as specified on the VIPAROUTE statement.

**System action**
TCP/IP continues, but the local stack cannot forward any DVIPA packets to the target stack.

**Operator response**
This might be a temporary condition that can resolve itself when the route becomes available. Use the Netstat ROUTe/-r command on the distributing stack to see possible routing failure problems to that target stack. If the problem cannot be resolved, contact the system programmer.

**System programmer response**
See the information about diagnosing OMPROUTE problems in *z/OS Communications Server: IP Diagnosis Guide* for information about routing failures.

**Module**
EZBXFDVI, EZBX6DVI, EZBXFMS4, EZBX6MS2

**EZD1172E** `tcpstackname` DETERMINED THAT ALL PARTNERS WERE UNREACHABLE FOR AT LEAST `timevalue` SECONDS

**Explanation**
Sysplex problem detection determined that there are no routes available to any partners.

`tcpstackname` is the name of the TCP/IP stack.

`timevalue` is the number of seconds that connectivity was not available.

**System action**
TCP/IP continues.

- If the GLOBALCONFIG SYSPLEXMONITOR RECOVERY option is active and this stack is not the only member of its TCP/IP sysplex group, the following RECOVERY actions will occur:
  - This stack will leave the TCP/IP sysplex group.
  - This stack will no longer participate in sysplex distribution (as a distributor or target) or act as an owner or a backup for DVIPAs. All DVIPAs defined on this stack will be deactivated; however, the DVIPA definitions will be saved.
If the problem is corrected, this operator message will be deleted; if the GLOBALCONFIG SYSPLEXMONITOR AUTOREJOIN option is active, the stack will process the DVIPA definitions and rejoin the TCP/IP sysplex group.

- If the GLOBALCONFIG SYSPLEXMONITOR NORECOVERY option is active, no action will be taken. If the problem is corrected, this operator message will be deleted.

See the information about sysplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide for more information.

See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for the definition of the SYSPLEXMONITOR parameters.

Operator response
Contact the system programmer.

System programmer response
Issue Netstat ROUTe/-r to determine which interfaces are used by the VIPAROUTE statements. Issue the Netstat DEVlinks/-d command to show which interfaces are active; check the system log for any messages related to the status of interfaces. If you cannot determine why the routes were lost, see the information about diagnosing OMPROUTE problems in z/OS Communications Server: IP Diagnosis Guide for additional information.

If this problem can be corrected, this operator message will be deleted.

If RECOVERY and NOAUTOREJOIN are active, then issue the VARY TCPIP,,SYSPLEX,JOINGROUP command to cause the DVIPA definitions to be processed, and the stack to rejoin the TCP/IP sysplex group.

If RECOVERY and AUTOREJOIN are active, no further actions are needed. The stack will process the DVIPA definitions and rejoin the TCP/IP sysplex group.

If NORECOVERY is active, no further actions are needed.

After connectivity is reestablished, if RECOVERY is active, wait until message EZD1172E is issued when the process of leaving the sysplex group completes successfully. Then issue the VARY TCPIP,, SYSPLEX,JOINGROUP command to cause the stack to rejoin the sysplex group. If NORECOVERY is active, no further actions are needed.

Module
EZBXFPDM

Procedure name
EZBXFPDM

EZD1173I VIPAROUTE IS NOT ENABLED FOR dxcf_address target_ipaddress - TARGET IP ADDRESS IS NOT VALID

Explanation
The target stack identified by dxcf_address is active, but target_ipaddress is not defined at that target stack or it is defined, but the address is not valid (for example, a dynamic VIPA address or loopback is used). The local stack will forward DVIPA packets to the target stack using dynamic XCF interfaces.

dxcf_address is the dynamic XCF address of a target stack as specified on the VIPAROUTE statement.
target_ipaddress is the IP address as specified on the VIPAROUTE statement.

System action
TCP/IP continues.
**Operator response**
Contact the system programmer.

**System programmer response**
Take the following actions to correct the problem:

- Ensure that the VIPAROUTE statement specifies the correct DXCF address and target IP address for the desired target stack.
- Ensure that `target_ipaddress` is correctly defined in the HOME list of the target stack and that it is an address that is valid for use as a target IP address.

See z/OS Communications Server: IP Configuration Reference for information about the VIPAROUTE statement.

**Module**
EZBXFDVI, EZBX6DVI, EZBXFMS4, EZBX6MS2

**Procedure name**
EZBXFDVI, EZBX6DVI, EZBXFMS4, EZBX6MS2

---

**EZD1174I**  
THE ROUTE FOR VIPAROUTE `dxcf_address target_ipaddress` IS NOW ACTIVE

**Explanation**
The route to `target_ipaddress` on the target stack identified by `dxcf_address` is now active. As a result, the local stack can forward DVIPA packets to the target stack.

`dxcf_address` is the dynamic XCF address of a target stack as specified on the VIPAROUTE statement.

`target_ipaddress` is the IP address in the HOME list of the target stack as specified on the VIPAROUTE statement.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZBXFHS, EZBX6HSH

**Procedure name**
Check_Route, Check_Route6

---

**EZD1175I**  
VARY TCPIP,,SYSPLEX,JOINGROUP COMMAND IGNORED BECAUSE `tcpstackname` IS ALREADY A MEMBER OF A TCP/IP SYSPLEX GROUP

**Explanation**
The command was ignored because the TCP/IP stack is already a member of a TCP/IP sysplex group.

`tcpstackname` is the name of the TCP/IP stack.
**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZBXFIO2

**Procedure name**
EZBXFSVS

---

**EZD1176I tcpstackname HAS SUCCESSFULLY JOINED THE TCP/IP SYSPLEX GROUP groupname**

**Explanations**
The TCP/IP stack successfully joined the TCP/IP sysplex group *groupname*. This stack can participate in sysplex distribution (as a distributor or target) or act as an owner or a backup for DVIPAs.

*tcpstackname* is the name of the TCP/IP stack.

*groupname* is the name of the TCP/IP sysplex group that was joined.

---

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZBXFINI

**Procedure name**
Join_Sysplex_Group

---

**EZD1177I tcpstackname SYSPLEX PROFILE DEFINITIONS ARE IGNORED BECAUSE THE STACK IS NO LONGER A MEMBER OF A TCP/IP SYSPLEX GROUP**

**Explanations**
The sysplex profile definitions in the data set referenced by the VARY TCPIP,OBEYFILE command were ignored because the stack is no longer a member of a TCP/IP sysplex group. Sysplex profile definitions cannot be applied if the stack is not a member of a TCP/IP sysplex group.

The stack left the TCP/IP sysplex group because one of the following occurred:

- A VARY TCPIP,,SYSPLEX,LEAVEGROUP command was issued, causing the stack to leave the TCP/IP sysplex group.
• A problem was detected that caused the stack to leave the TCP/IP sysplex group.

\textit{tcpstackname} is the name of the TCP/IP stack.

\textbf{System action}

TCP/IP continues.

\textbf{Operator response}

If a VARY TCPIP,,SYSPLEX,LEAVEGROUP command was issued, then issue a VARY TCPIP,,SYSPLEX,JOINGROUP command to cause the stack to rejoin the TCP/IP sysplex group. If a problem was detected, prior eventual action messages explain why the stack is not a member. The problem that caused the stack to leave the TCP/IP sysplex group should be identified and corrected so that the stack can rejoin the TCP/IP sysplex group. See the information about sysplex problem detection and recovery in \textit{z/OS Communications Server: IP Configuration Guide} for more information. If the problem cannot be corrected, contact the system programmer. When the stack has rejoined the TCP/IP sysplex group, the sysplex profile definitions can be reapplied.

\textbf{System programmer response}

If the problem causing the stack to leave the TCP/IP sysplex group cannot be corrected, contact IBM software support services.

\textbf{Module}

EZBXFDYN

\textbf{Procedure name}

EZBXFDYN

\textbf{EZD1178I} \hspace{1em} THE VARY TCPIP,,SYSPLEX,JOINGROUP COMMAND WAS ACCEPTED

\textbf{Explanation}

A VARY TCPIP,,SYSPLEX,JOINGROUP command was accepted. Later messages will indicate whether the stack successfully joined a TCP/IP sysplex group.

\textbf{System action}

TCP/IP continues.

\textbf{Operator response}

None.

\textbf{System programmer response}

None.

\textbf{Module}

EZBXFIO2

\textbf{Procedure name}

EZBXFSVS

\textbf{EZD1179I} \hspace{1em} VIPAROUTE DEFINE REJECTED - dxcf\_address IS ALREADY DEFINED
Explanation
The VIPAROUTE DEFINE statement was rejected. The same dynamic XCF address was already defined with a different target IP address on a prior VIPAROUTE DEFINE statement.

dxcf_address is the dynamic XCF address that was specified on the VIPAROUTE DEFINE statement that was rejected.

System action
TCP/IP continues but the VIPAROUTE DEFINE statement is ignored.

Operator response
Contact the system programmer.

System programmer response
Correct the VIPAROUTE statement and issue a VARY TCPIP,,OBEYFILE command with the new VIPADYNAMIC block.

Module
EZBXFDVI, EZBX6DVI

Procedure name
EZBXFDVI, EZBX6DVI

EZD1180I  VIPAROUTE DELETE REJECTED - dxcf_address target_ipaddress IS NOT DEFINED

Explanation
The VIPAROUTE DELETE statement was rejected. The dxcf_address with target_ipaddress was not found on a previously defined VIPAROUTE DEFINE statement.

dxcf_address is the dynamic XCF address that was specified on the VIPAROUTE DELETE statement that was rejected.

target_ipaddress is the IP address in the HOME list of the target stack as specified on the VIPAROUTE statement.

System action
TCP/IP continues but the VIPAROUTE DELETE statement is ignored.

Operator response
Contact the system programmer.

System programmer response
Correct the VIPAROUTE DELETE statement and issue a VARY TCPIP,,OBEYFILE command with a new VIPADYNAMIC block.

Module
EZBXFDVI, EZBX6DVI

Procedure name
EZBXFDVI, EZBX6DVI

EZD1181I  VIPAROUTE DEFINE dxcf_address REJECTED - ENTRIES EXCEEDED
Explanation
The VIPAROUTE DEFINE statement was rejected. The maximum number of 256 VIPAROUTE entries was exceeded.

dxcf_address is the dynamic XCF address that was coded on the VIPAROUTE statement that was rejected.

System action
TCP/IP continues but the VIPAROUTE statement is ignored.

Operator response
Contact the system programmer.

System programmer response
Issue a VARY TCPIP,OBEYFILE command with a new VIPADYNAMIC block to remove unneeded VIPAROUTE entries before adding new ones.

Module
EZBXFDVI, EZBX6DVI

Procedure name
EZBXFDVI, EZBX6DVI

EZD1182I TARGET SERVER FOR dvipa_or_intfname PORT portnum AT tcp_name ON host_name IS NOT RESPONSIVE - TCSR tcsr_value CER cer_value SEF sef_value

Explanation
The server that is servicing the specified distributed dynamic virtual IP address (DVIPA) and port number became unresponsive while it was processing connection setup requests. The server is no longer successfully processing incoming connection requests.

In the message text:

**dvipa_or_intfname**
- For IPv4, specifies the distributed DVIPA that is being serviced by the target server.
- For IPv6, specifies the interface name associated with the distributed DVIPA that is being serviced by the target server.

**portnum**
The port that is being serviced by the target server.

**tcp_name**
The name of the TCP/IP stack that the target server is using.

**host_name**
The name of the MVS system that the TCP/IP stack is running on.

**tcsr_value**
The percent of connection setup requests that are routed from the distributor and that are successfully received by the target for this server.

**cer_value**
The percentage of connection setup requests that are received by the target for this server and that achieve the connection established state.

**sef_value**
The server efficiency factor for the server application. This value is an indication of how effectively the application is accepting new connection requests and managing its backlog queue.
System action
Processing continues. New connection setup requests are not routed to this target server until the target server recovers. Message EZD1183I will be issued at that time.

Operator response
See the information about diagnosing sysplex problems in z/OS Communications Server: IP Diagnosis Guide for more information.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Sysplex Distributor

Module
EZBXFWLM

Routing code
10

Descriptor code
12

Automation
This message is sent to the system console. Although this target server is no longer an eligible target server, you might want to review why your target server is not performing correctly.

Example
EZD1182I  TARGET SERVER FOR 192.3.10.13 PORT 23 AT TCPCS ON MVS128 IS NOT RESPONSIVE - TCSR 100 CER 0 SEF 80

EZD1183I  TARGET SERVER FOR dvipa_or_intfname PORT portnum AT tcp_name ON host_name IS NOW RESPONSIVE

Explanation
The server servicing the specified distributed DVIPA and port number resumed responsiveness in processing connection setup requests. The server is now successfully processing incoming connection requests.

dvipa_or_intfname:
- For IPv4, this is the distributed dynamic virtual IP address that is being serviced by the target server.
- For IPv6, this is the interface name associated with the dynamic virtual IP address that is being serviced by the target server.
portnum is the port that is being serviced by the target server.
tcp_name is the name of the TCP/IP stack that the target server is using.
host_name is the name of the MVS system that the TCP/IP stack is running on.

**System action**
Processing continues. New connection setup requests will now be routed to this target server.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZBXFWLM

**Procedure name**
EZBXFPWS

**EZD1185I** THE VARY TCPIP,,SYSPLEX,DEACTIVATE, DVIPA COMMAND WAS IGNORED BECAUSE THE DVIPA IS ALREADY DEACTIVATED

**Explanation**
A VARY TCPIP,,SYSPLEX,DEACTIVATE, DVIPA command was issued for a DVIPA. The command was ignored because the DVIPA is currently deactivated. The Netstat VIPADCFG/-F command can be used to see all of the currently active DVIPAs and deactivated DVIPAs.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZBXFIO2

**Procedure name**
EZBXFSVS

**EZD1186I** THE VARY TCPIP,,SYSPLEX,DEACTIVATE,DVIPA COMMAND WAS REJECTED BECAUSE THE DVIPA IS NOT DEFINED BY VIPADEFINE OR VIPABACKUP ON THIS STACK

**Explanation**
A VARY TCPIP,,SYSPLEX,DEACTIVATE,DVIPA command was issued for a DVIPA. The command was rejected because the DVIPA is not defined by a VIPADEFINE or VIPABACKUP statement on this stack. The VARY
TCP/IP, SYSPLEX, DEACTIVATE, DVIPA command can be issued only on the stack that has a VIPADEFINE or VIPABACKUP definition for this DVIPA. Use the Netstat VIPADCFG/-F command to see all the currently active DVIPAs and deactivated DVIPAs.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZBXFIO2

**Procedure name**
EZBXFSVS

**EZD1187E tcpstackname WAS NOT ABLE TO GET TCP/IP storagetype STORAGE**

**Explanation**
As a result of storage limits set by GLOBALCONFIG, tcpstackname was not able to get TCP/IP storagetype storage.

tcpstackname is the name of the TCP/IP stack.

storagetype is the type of storage that was unavailable. storagetype will be either **PRIVATE** or **ECSA** (extended common storage area).

**System action**
TCP/IP continues.

- If the GLOBALCONFIG SYSPLEXMONITOR RECOVERY option is active and this stack is not the only member of its TCP/IP sysplex group, the following RECOVERY actions will occur:
  - This stack will leave the TCP/IP sysplex group.
  - This stack will no longer participate in sysplex distribution (as a distributor or target) or act as an owner or a backup for DVIPAs. All DVIPAs defined on this stack will be deleted; however, the DVIPA definitions will be saved.
  - If the problem is corrected, this operator message will be deleted; if the GLOBALCONFIG SYSPLEXMONITOR AUTOREJOIN option is active, the stack will process the saved DVIPA definitions and rejoin the TCP/IP sysplex group.
- If the GLOBALCONFIG SYSPLEXMONITOR NORECOVERY option is active, no action will be taken. If the problem is corrected, this operator message will be deleted.

See the information about sysplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide. See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for the definitions of the SYSPLEXMONITOR parameters.

**Operator response**
If the storage problem cannot be corrected, save the TCP/IP profile and system log. If a dump was not created, then take a dump of the TCP/IP address space and dataspaces. Contact the system programmer.
**System programmer response**

Use the display TCP/IP storage command `D TCPIP,,STOR` to determine the current status of TCP/IP PRIVATE and ECSA storage. See the information about the `DISPLAY TCPIP,,STOR` command in z/OS Communications Server: IP System Administrator's Commands for more information about using this command.

Storage limits were set using the GLOBALCONFIG ECSALIMIT or GLOBALCONFIG POOLLIMIT statement. It might be necessary to raise these limits to correct this problem; look for earlier TCP/IP warning messages that are issued each time a storage limit boundary is crossed (OK, CONSTRAINED, CRITICAL, and EXHAUSTED).

See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for the definitions of the ECSALIMIT and POOLLIMIT. For ECSA storage exhaustion, it is necessary to determine which jobs or address spaces are using an excessive amount of storage. To determine the users of ECSA storage, common storage tracking should be enabled. See the z/OS MVS Initialization and Tuning Guide for information about requesting common storage tracking. Use the VERBEXIT VSMDATA OWNCOMM SUMMARY command to determine how much storage is used by each job. See the z/OS MVS Diagnosis: Tools and Service Aids for information about the IPCS VERBEXIT VSMDATA command.

If the storage problem cannot be corrected, contact IBM software support services with all supporting documentation.

If this problem can be corrected, this operator message will be deleted.

If RECOVERY and NOAUTOREJOIN are active, then issue the `VARY TCPIP,,SYSPLEX,JOINGROUP` command to cause the saved DVIPA definitions to be processed, and the stack to rejoin the TCP/IP sysplex group.

If RECOVERY and AUTOREJOIN are active, no further actions are needed. The stack will process the DVIPA definitions and rejoin the TCP/IP sysplex group.

If NORECOVERY is active, no further actions are needed.

**Module**

EZBXFPDM

**Procedure name**

EZBXFPDM

**EZD1188I** THE VARY TCPIP,,SYSPLEX,REACTIVATE,DVIPA COMMAND FAILED

**Explanation**

A VARY TCPIP,,SYSPLEX,REACTIVATE,DVIPA command was issued for a DVIPA. The command failed. A prior error message will indicate why the command failed.

**System action**

TCP/IP continues. The specified DVIPA definition remains deactivated.

**Operator response**

Use the prior error message to determine why the command failed.

**System programmer response**

None.

**Module**

EZBXFDVI, EZBX6DVI
**Procedure name**
ValidateVDEF,ValidateVBKKUP

**EZD1189I**   THE VARY TCPIP,,SYSPLEX,REACTIVATE,DVIPA COMMAND COMPLETED SUCCESSFULLY

**Explanation**
A VARY TCPIP,,SYSPLEX,REACTIVATE,DVIPA command completed successfully for a DVIPA.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZBXFDVI,EZBX6DVI

**Procedure name**
Add_DVIPAentry,Add_DVIPAentry6

**EZD1190I**   THE VARY TCPIP,,SYSPLEX,LEAVEGROUP COMMAND WAS IGNORED BECAUSE THE STACK IS NOT A MEMBER OF A TCP/IP SYSPLEX GROUP

**Explanation**
The VARY TCPIP,,SYSPLEX,LEAVEGROUP command was ignored because the stack is not a member of a TCP/IP sysplex group.

**System action**
TCP/IP continues.

**Operator response**
See the information about sysplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide for more information.

**System programmer response**
None.

**Module**
EZBXFIO2

**Procedure name**
EZBXFSVS

**EZD1191I**   THE VARY TCPIP,,SYSPLEX,JOINGROUP COMMAND WAS IGNORED BECAUSE SYSPLEX PROBLEM DETECTION CLEANUP HAS NOT FINISHED
Explanation
A problem detected by Sysplex Autonomics caused the stack to leave the TCP/IP sysplex group and cleanup all DVIPA resources. The DVIPA resource cleanup has not completed.

System action
TCP/IP continues. One of the following two messages will be issued when the problem detection cleanup process completes:
- EZZ9675E to indicate that the resource cleanup process failed.
- EZZ9676E to indicate that the resource cleanup process has successfully completed.

Operator response
Wait until either EZZ9675E or EZZ9676E is issued. If message EZZ9675E is issued, restart the stack to rejoin the TCP/IP sysplex group. If message EZZ9676E is issued and the stack does not automatically rejoin the sysplex group, see message EZZ9676E in z/OS Communications Server: IP Messages Volume 4 (EZZ, SNM) for more information about how to cause the stack to rejoin the sysplex group. See the information about sysplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide for more information.

System programmer response
None.

Module
EZBXFIO2

Procedure name
EZBXFSVS

EZD1192I THE VIPADYNAMIC CONFIGURATION WAS SUCCESSFULLY RESTORED FOR tcpstackname

Explanation
The stack rejoined the TCP/IP sysplex group and all its VIPADYNAMIC configuration definitions were successfully restored.
tcpstackname is the name of the TCP/IP stack.

System action
TCP/IP continues

Operator response
None.

System programmer response
None.

Module
EZBXFDYN

Procedure name
EZBXFDYN
EZD1193I  ALL OF THE VIPADYNAMIC CONFIGURATION DEFINITIONS FOR tcpstackname COULD NOT BE RESTORED

Explanation
The stack rejoined the TCP/IP sysplex group, but one or more of the stack VIPADYNAMIC definitions could not be restored because of a conflict with existing definitions on this or another stack in the TCP/IP sysplex group.

tcpstackname is the name of the TCP/IP stack.

System action
TCP/IP continues. VIPADYNAMIC configuration definitions that could not be restored were deleted. Configuration definitions that did not have a conflict were activated.

Operator response
See prior error messages to determine which configuration definitions were not restored. If you want to configure these definitions on this stack, remove the conflicting configuration (on this or another stack) and invoke the VARY TCPIP,,OBEYFILE command, referencing a data set that contains the rejected VIPADYNAMIC definitions.

System programmer response
None.

Module
EZBXFDYN

Procedure name
EZBXFDYN

EZD1194E  tcpstackname SYSPLEX PROCESSING ENCOUNTERED A NONRECOVERABLE ERROR WHILE TRYING TO RESTORE THE SAVED SYSPLEX CONFIGURATION

Explanation
A nonrecoverable error was encountered while attempting to restore the saved configuration and join a TCP/IP sysplex group.

tcpstackname is the name of the TCP/IP stack.

System action
TCP/IP continues. The stack must be restarted to join a TCP/IP sysplex group. If AUTOREJOIN was configured, it is now disabled.

Operator response
Save the TCP/IP profile and system log. If a dump was not created, then take a dump of the TCP/IP address space and dataspaces. Contact the system programmer.

System programmer response
Contact IBM software support services with the TCP/IP profile, system log and dump.

Module
EZBXFUT4
EZBXFRSC

**EZD1195I**  THE VARY TCPIP,,SYSPLEX,JOINGROUP COMMAND WAS REJECTED. THE STACK MUST BE RESTARTED TO JOIN A TCP/IP SYSPLEX GROUP

**Explanation**
The VARY SYSPLEX,JOINGROUP command was rejected because either sysplex problem detection cleanup failed as the stack left the TCP/IP sysplex group, or a previous attempt to process the saved sysplex configuration and join a TCP/IP sysplex group failed. See the explanation of message EZD1194E (issued if failure occurs while processing the saved sysplex configuration) or EZZ9675E (issued for sysplex cleanup failure) in z/OS Communications Server: IP Messages Volume 4 (EZZ, SNM) for more information.

**System action**
TCP/IP continues. The stack must be restarted to join a TCP/IP sysplex group.

**Operator response**
See the explanations of messages EZD1194E or EZZ9675E for more information.

**System programmer response**
See the explanations of messages EZD1194E or EZZ9775E for more information.

**Module**
EZBXFSVS

EZBXFIO2

**EZD1196I**  THE VARY TCPIP,,SYSPLEX,REACTIVATE,DVIPA COMMAND WAS IGNORED BECAUSE THE DVIPA IS NOT DEACTIVATED

**Explanation**
A VARY TCPIP,,SYSPLEX,REACTIVATE, DVIPA command was issued for a DVIPA. The command was ignored because the DVIPA is not deactivated or it does not exist on this stack. Use the Netstat VIPADCFG/-F command to see all of the currently active DVIPAs and deactivated DVIPAs.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZBXFIO2
EZBXFSVS

EZD1197I  THE VARY TCPIP,,SYSPLEX,DEACTIVATE,DVIPA COMMAND COMPLETED SUCCESSFULLY

Explanation
A VARY TCPIP,,SYSPLEX,DEACTIVATE,DVIPA command successfully completed for a DVIPA.

System action
TCP/IP continues.

Operator response
None.

System programmer response
None.

Module
EZBXFUT4

Procedure name
EZBXFVVS

EZD1198I  THE VARY TCPIP,,SYSPLEX,cmdtype COMMAND COMPLETED SUCCESSFULLY

Explanation
A VARY SYSPLEX QUIESCE or RESUME command completed successfully.

System action
TCP/IP continues.

Operator response
None.

System programmer response
None.

Module
EZBXFUT4

Procedure name
Process_QUIESCE_cmd, Process_RESUME_cmd

EZD1199I  THE VARY TCPIP,,SYSPLEX,cmdtype COMMAND WAS IGNORED BECAUSE NO APPLICATION
WAS FOUND LISTENING WITH THE SPECIFIED COMMAND PARAMETERS
Explanation

A VARY SYSPLEX QUIESCE or RESUME command was issued for a specific port. The command was ignored because no application was found listening with the parameters specified on the command.

`cmdtype` is either QUIESCE or RESUME.

System action

TCP/IP continues.

Operator response

Reissue the command after correcting the command parameters. See the information about the VARY QUIESCE command and the VARY RESUME command in z/OS Communications Server: IP System Administrator’s Commands for more information.

System programmer response

None.

Module

EZBXFUT4

Procedure name

Process_QUIESCE_cmd, Process_RESUME_cmd

EZD1200I  THE VARY TCPIP,,SYSPLEX,cmdtype COMMAND WAS REJECTED BECAUSE MORE THAN ONE LISTENING APPLICATION WAS FOUND MATCHING THE COMMAND PARAMETERS

Explanation

A VARY SYSPLEX QUIESCE or RESUME command was issued for a specific port. The command was rejected because there is more than one listening application matching the parameters specified on the command.

`cmdtype` is either QUIESCE or RESUME.

System action

TCP/IP continues.

Operator response

You must specify the JOBNAME and possibly the ASID parameters to identify a unique listening application. See the information about the VARY QUIESCE command and the VARY RESUME command in z/OS Communications Server: IP System Administrator’s Commands for more information.

System programmer response

None.

Module

EZBXFUT4

Procedure name

Process_QUIESCE_cmd, Process_RESUME_cmd

EZD1201I  THE VARY TCPIP,,SYSPLEX,RESUME,PORT COMMAND WAS REJECTED BECAUSE IT FOLLOWS A VARY TCPIP,,SYSPLEX,QUIESCE,TARGET COMMAND
**Explanation**

A VARY TCPIP,,SYSPLEX,RESUME,PORT command cannot follow a VARY TCPIP,,SYSPLEX,QUIESCE,TARGET command. The only valid RESUME command that can follow a VARY TCPIP,,SYSPLEX,QUIESCE,TARGET is VARY TCPIP,,SYSPLEX,RESUME,TARGET.

**System action**

TCP/IP continues.

**Operator response**

You must specify VARY TCPIP,,SYSPLEX,RESUME,TARGET to resume the listening applications for DVIPA sysplex distributor workload distribution following a VARY TCPIP,,SYSPLEX,QUIESCE,TARGET command.

**System programmer response**

None.

**Module**

EZBXFUT4

**Procedure name**

Process_RESUME_cmd

---

**EZD1202I**

THE VARY TCPIP,,SYSPLEX,RESUME,TARGET COMMAND WAS REJECTED BECAUSE IT WAS NOT PRECEDED BY A CORRESPONDING VARY TCPIP,,SYSPLEX,QUIESCE,TARGET COMMAND

**Explanation**

A VARY TCPIP,,SYSPLEX,RESUME,TARGET command can only follow a VARY TCPIP,,SYSPLEX,QUIESCE,TARGET command.

**System action**

TCP/IP continues.

**Operator response**

You must specify VARY TCPIP,,SYSPLEX,RESUME,PORT to resume a listening application for DVIPA sysplex distributor workload distribution that has been previously quiesced using VARY TCPIP,,SYSPLEX,QUIESCE,PORT command.

**System programmer response**

None.

**Module**

EZBXFUT4

**Procedure name**

Process_RESUME_cmd

---

**EZD1203I**

VIPADISTRIBUTE NOT ALLOWED BECAUSE dvipa_or_intfname IS CURRENTLY DEACTIVATED
Explanation
A VIPADISTRIBUTE DEFINE or DELETE statement appeared in the file referenced by a VARY TCPIP, OBEYFILE command, but the specified dynamic virtual P address (DVIPA) has been deactivated and distribution cannot be added or changed while the DVIPA is deactivated.

dvipa_or_intfname:
- For IPv4, this is the DVIPA that is specified on the VIPADISTRIBUTE statement.
- For IPv6, this is the interface name that is specified on the VIPADISTRIBUTE statement.

System action
TCP/IP continues. The specified DVIPA remains deactivated with its distribution definitions unchanged.

Operator response
Contact the system programmer.

System programmer response
If you want to change the distribution for this DVIPA before reactivating it, you must issue a VARY TCPIP, OBEYFILE command referencing a file containing a VIPADELETE statement for the DVIPA. Because the DVIPA is deactivated, this will delete both the DVIPA and its distribution definitions. To redefine the DVIPA on this stack, issue a VIPADEFINE or VIPABACKUP statement, followed by any VIPADISTRIBUTE statements, for the DVIPA. This can be done with the same, or a subsequent, VARY TCPIP, OBEYFILE command.

Module
EZBXFDVI, EZBX6DVI

Procedure name
ValidateVDIST, ValidateVDIST6

EZD1204I DYNAMIC VIPA dvipa WAS CREATED USING IOCTL BY jobname ON tcpstackname

Explanation
The application instance DVIPA specified by the dvipa value was created dynamically by executing the MODDVIPA utility or by an application invoking the SIOCSVIPA or SIOCSVIPA6 IOCTL.

In the message text:

dvipa
- The dynamic VIPA that was created.

jobname
- The job name of either the application or the MODDVIPA utility that issued the SIOCSVIPA or SIOCSVIPA6 IOCTL.

tcpstackname
- The name of the TCP/IP stack.

See the information about configuring the unique application-instance scenario in z/OS Communications Server: IP Configuration Guide for more information.

System action
TCP/IP continues.

Operator response
None.
**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Module**
EZBXFUDV

**Example**
None.

**EZD1205I**  DYNAMIC VIPA $dvipa$ WAS CREATED USING BIND BY $jobname$ ON $tcpstackname$

**Explanation**
The application instance DVIPA specified by the $dvipa$ value was created dynamically using a BIND.

In the message text:
- $dvipa$  The dynamic VIPA that was created.
- $jobname$  The job name of the application that issued the BIND.
- $tcpstackname$  The name of the TCP/IP stack.

See the information about configuring the unique application-instance scenario in z/OS Communications Server: IP Configuration Guide for more information.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Module**
EZBXFUDV
Example
None.

**EZD1206I**  DYNAMIC VIPA *dvipa* WAS DELETED USING IOCTL BY *jobname* ON *tcpstackname*

**Explanation**
The specified DVIPA was deleted dynamically by executing the MODDVIPA utility or by an application invoking the SIOCSVIPA or SIOCSVIPA6 IOCTL.

In the message text:

**dvipa**  
The dynamic VIPA that was deleted.

**jobname**  
The job name of either the application or the MODDVIPA utility that issued the SIOCSVIPA or SIOCSVIPA6 IOCTL.

**tcpstackname**  
The name of the TCP/IP stack.

See the information about configuring the unique application-instance scenario in z/OS Communications Server: IP Configuration Guide for more information.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Module**
EZBXFUDV

**Example**
None.

**EZD1207I**  DYNAMIC VIPA *dvipa* WAS DELETED USING CLOSE API BY *jobname* ON *tcpstackname*

**Explanation**
The specified application instance DVIPA was deleted dynamically by a CLOSE API.

In the message text:

**dvipa**  
The dynamic VIPA that was deleted.
**jobname**
The job name of the application that issued the CLOSE API.

**tcpstackname**
The name of the TCP/IP stack.

See the information about configuring the unique application-instance scenario in z/OS Communications Server: IP Configuration Guide for more information.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Module**
EZBXFUDV

**Example**
None.

---

**EZD1208I**  VIPADISTRIBUTE WITH BOTH TIMEDAFFINITY AND OPTLOCAL REJECTED

**Explanation**
Both the OPTLOCAL keyword and the TIMEDAFFINITY keyword were specified on the same VIPADISTRIBUTE statement. The OPTLOCAL and TIMEDAFFINITY keywords are mutually exclusive on the VIPADISTRIBUTE statement.

**System action**
TCP/IP continues. The VIPADISTRIBUTE statement is rejected.

**Operator response**
Contact the system programmer.

**System programmer response**
To change the VIPADISTRIBUTE statement to have only one keyword, do one of the following:
- Correct and resubmit the original profile statement with the new VIPADYNAMIC block.
- Issue the VARY TCPIP,,OBEYFILE command with the new VIPADYNAMIC block.

See the information about the VIPADYNAMIC command in z/OS Communications Server: IP Configuration Reference for more information.
User response
Not applicable.

Problem determination
Not applicable.

Module
EZBXFDVI, EZBX6DVI

Example
None.

**EZD1209E tcpstackname DETERMINED THAT ALL MONITORED INTERFACES WERE NOT ACTIVE FOR AT LEAST timevalue SECONDS**

**Explanation**
Sysplex problem detection has determined that all monitored interfaces are inactive.

In the message text:
- **tcpstackname**
  - The name of the TCP/IP stack.
- **timevalue**
  - The number of seconds during which all monitored interfaces were not active.

**System action**
TCP/IP continues.

- If the GLOBALCONFIG SYSPLEXMONITOR RECOVERY option is active and this stack is not the only member of its TCP/IP sysplex group, the following RECOVERY actions occur:
  - This stack leaves the TCP/IP sysplex group.
  - This stack no longer participates in sysplex distribution (as a distributor or target) nor acts as an owner or a backup for DVIPAs. All DVIPAs defined on this stack are deactivated; however, the DVIPA definitions are saved.
  - If the problem is corrected, this operator message is deleted; if the GLOBALCONFIG SYSPLEXMONITOR AUTOREJOIN option is active, the stack processes the DVIPA definitions and rejoins the TCP/IP sysplex group.
- If the GLOBALCONFIG SYSPLEXMONITOR NORECOVERY option is active, no action is taken. If the problem is corrected, this operator message is deleted.

See the information about network interfaces monitoring and sysplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide for more information.

See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for the definitions of the SYSPLEXMONITOR parameters.

**Operator response**
Contact the system programmer.

**System programmer response**
Issue the Netstat DEvlinks/-d command to determine which interfaces are being monitored; check the system log for any messages related to the status of monitored interfaces. If the monitored interfaces are inactive as a result of stopping devices or interfaces, disable the monitoring of these interfaces by specifying the
NOMONSYSPLEX keyword on the LINK and INTERFACE statement through the VARY TCPIP,,OBEYFILE command before stopping the devices or interfaces. See the information about the DEVICE and LINK statements and the INTERFACE statements in z/OS Communications Server: IP Configuration Reference for more information. If you cannot determine why the monitored interfaces are inactive, contact IBM software support services after obtaining the system log and the TCPIP profile.

If this problem can be corrected, this operator message is deleted.

- If RECOVERY and NOAUTOREJOIN are active, then issue the VARY TCPIP,,SYSPLEX,JOINGROUP command to cause the DVIPA definitions to be processed, and cause the stack to rejoin the TCP/IP sysplex group.
- If RECOVERY and AUTOREJOIN are active, no further actions are needed. The stack processes the DVIPA definitions and rejoins the TCP/IP sysplex group.
- If NORECOVERY is active, no further actions are needed.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Module**
EZBXFPDM

**Example**
None.

**EZD1210E tcpstackname DETERMINED THAT NO DYNAMIC ROUTES OVER MONITORED INTERFACES WERE FOUND FOR AT LEAST timeval SECONDS**

**Explanation**
Sysplex problem detection has determined that no dynamic routes over monitored interfaces were found. In the message text:

*tcpstackname*  
The name of the TCP/IP stack.

*timevalue*  
The number of seconds during which no dynamic routes over monitored interfaces were found.

**System action**
TCP/IP continues.

- If the GLOBALCONFIG SYSPLEXMONITOR RECOVERY option is active and this stack is not the only member of its TCP/IP sysplex group, the following RECOVERY actions occur:
  - This stack leaves the TCP/IP sysplex group.
  - This stack no longer participates in sysplex distribution (as a distributor or target) nor acts as an owner or a backup for DVIPAs. All DVIPAs defined on this stack are deactivated; however, the DVIPA definitions are saved.
  - If the problem is corrected, this operator message is deleted; if the GLOBALCONFIG SYSPLEXMONITOR AUTOREJOIN option is active, the stack processes the DVIPA definitions and rejoins the TCP/IP sysplex group.
- If the GLOBALCONFIG SYSPLEXMONITOR NORECOVERY option is active, no action is taken. If the problem is corrected, this operator message is deleted.
See the information about network interfaces monitoring and sysplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide for more information.

See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for the definitions of the SYSPLEXMONITOR parameters.

**Operator response**

Contact the system programmer.

**System programmer response**

Issue the Netstat DEVlinks/-d command to determine which interfaces are being monitored; this recovery action might be triggered because the first hop routers were brought down for maintenance. You can temporarily disable the monitoring of dynamic routes by specifying MONINTERFACE NODYNROUTE on the GLOBALCONFIG SYSPLEXMONITOR statement through the VARY TCPIP,,OBEYFILE command. If you still cannot determine why the routes were lost, see the information about diagnosing OMPROUTE problems in z/OS Communications Server: IP Diagnosis Guide for more information.

If this problem can be corrected, this operator message is deleted.

- If RECOVERY and NOAUTOREJOIN are active, then issue the VARY TCPIP,,SYSPLEX,JOINGROUP command to cause the DVIPA definitions to be processed, and cause the stack to rejoin the TCP/IP sysplex group.
- If RECOVERY and AUTOREJOIN are active, no further actions are needed. The stack processes the DVIPA definitions and rejoins the TCP/IP sysplex group.
- If NORECOVERY is active, no further actions are needed.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Module**

EZBXFPDM

**Example**

None.

**EZD1211E tcpstackname DELAYING SYSPLEX PROFILE PROCESSING - ALL MONITORED INTERFACES WERE NOT ACTIVE**

**Explanation**

The TCP/IP stack delayed joining a sysplex group and delayed processing sysplex definitions in the profile (VIPADYNAMIC and IPCONFIG/IPCONFIG6 DYNAMICXCF statements).

In the message text:

- **tcpstackname** The name of the TCP/IP stack.

**System action**

TCP/IP continues.
Operator response

If you want the TCP/IP stack to immediately join a sysplex group rather than wait for monitored interfaces to be active, issue the VARY TCPIP,OBEYFILE command with GLOBALCONFIG SYSPLEXMONITOR NOMONINTERFACE specified. See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for information about the MONINTERFACE keyword.

When at least one monitored interface becomes active, TCP/IP joins a sysplex group and finishes processing the sysplex definitions.

System programmer response

None.

User response

Not applicable.

Problem determination

Not applicable.

Module

EZBXFDYN

Example

None.

EZD1212E tcpstackname DELAYING SYSPLEX PROFILE PROCESSING - NO DYNAMIC ROUTES OVER MONITORED INTERFACES WERE FOUND

Explanation

The TCP/IP stack delayed joining a sysplex group and delayed processing sysplex definitions in the profile (VIPADYNAMIC and IPCONFIG/IPCONFIG6 DYNAMICXCF statements).

In the message text:

tcpstackname
The name of the TCP/IP stack.

System action

TCP/IP continues.

Operator response

If you want the TCP/IP stack to immediately join a sysplex group rather than wait for dynamic routes over monitored interfaces to be found, issue the VARY TCPIP,OBEYFILE command with the GLOBALCONFIG SYSPLEXMONITOR NOMONINTERFACE option or the NODYNROUTE option specified. See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for information about the MONINTERFACE and DYNROUTE keywords.

When at least one dynamic route over monitored interfaces is found, TCP/IP joins a sysplex group and finishes processing the sysplex definitions.

If OMPROUTE is active, and the expected dynamic routes over the monitored interfaces are not being generated, contact the system programmer.
**System programmer response**

Issue the Netstat DEVlinks/-d command to determine which interfaces are being monitored. If the first hop routers were stopped for maintenance, this might be the reason why dynamic routes are not being generated. If you cannot determine why the routes are not being created, see the information about diagnosing OMPROUTE problems in z/OS Communications Server: IP Diagnosis Guide.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Module**
EZBXFDYN

**Example**
None.

**EZD1213I**  
A DISTMETHOD of **distribution_method** AND AN OPTLOCAL VALUE OTHER THAN ZERO ARE NOT COMPATIBLE ON A VIPADISTRIBUTE STATEMENT - THE OPTLOCAL VALUE IS SET TO ZERO

**Explanation**
The OPTLOCAL values in the range 1–16 are meaningful only when the specified distribution method (DISTMETHOD) value is either BASEWLM or SERVERWLM. For any other DISTMETHOD value, the only OPTLOCAL value allowed is 0.

In the message text:

**distribution_method**

The configured distribution method that is causing the OPTLOCAL value to be changed from its configured setting to a value of 0.

**System action**
The OPTLOCAL value is changed to 0. Processing of the VIPADISTRIBUTE statement continues.

**Operator response**
None.

**System programmer response**
To avoid receiving this informational message in the future, update the VIPADISTRIBUTE statement to specify an OPTLOCAL value of 0 or a distribution method of BASEWLM or SERVERWLM.

**User response**
None.

**Problem determination**
Not applicable.
**Module**
EZBXFDV2, EZBX6DV2

**Example**
None.

**EZD1214I**  INITIAL DYNAMIC VIPA PROCESSING HAS COMPLETED FOR tcpipstackname

**Explanation**
This message is issued when the first of the following occurs:

- At least one statement in a VIPADYNAMIC block in a profile/obeyfile has processed.
- This stack has been made a target by another stack in the sysplex.

This message is also issued when the stack has rejoined the sysplex and configuration processing has completed.

In the message text:

`tcpipstackname`

The name of the TCPIP stack.

**System action**
Processing continues.

**System programmer response**
None.

**User response**
None.

**Module**
EZBXFDYN

**Routing code**
2, 8

**Descriptor code**
12

**Automation**
This message is written to the console. Automation can use this message to start applications which require Dynamic VIPA addresses to be available or which will create Dynamic VIPA addresses.

**Procedure name**
mainline

**EZD1215I**  MLSCHK STARTED STACK tcpjobname USER stkuser SECLABEL stksl
Explanation
This message is written the first time the multilevel security consistency check is performed on a stack. It is also written each time the stack determines that the system has returned to MLACTIVE after being changed to NOMLACTIVE. When running on an MLACTIVE system, TCPIP performs a multilevel security consistency check in any of the following situations:

- After initial profile processing.
- After every VARY TCPIP, OBEYFILE command.
- After SYSPLEX dynamic VIPA changes.
- When it receives an ENF signal from RACF that indicates that a RACLST REFRESH was done for the SERVAUTH or SECLABEL class.

 tcpjobname is the name of the TCPIP job.
 stkuser is the user ID of this TCPIP job.
 stksl is the security label of this TCPIP job, or <NONE> if the job was started without a security label.

System action
Processing continues.

Operator response
None.

System programmer response
None. See z/OS Communications Server: IP Diagnosis Guide for information about how to set the user and seclabel values.

Module
EZBTIMDF

Procedure name
MLSCheck

EZD1216I  MLSCHK SUCCEEDED STACK tcpjobname IS MULTILEVEL SECURE

Explanation
When running on an MLACTIVE system, TCPIP performs a multilevel security consistency check in any of the following situations:

- After initial profile processing.
- After every VARY TCPIP, OBEYFILE command.
- After SYSPLEX dynamic VIPA changes.
- When it receives an ENF signal from RACF that indicates that a RACLST REFRESH was done for the SERVAUTH or SECLABEL class.

No configuration inconsistencies were found. This message is issued the first time multilevel security checking completes successfully after the stack is started or the first time it completes successfully after a failure.

tcpjobname is the job TCPIP JOBNAME.

System action
Processing continues.
Operator response
None.

System programmer response
None.

Module
EZBTIMDF

Procedure name
MLSCheck

EZD1217I  MLSCHK FAILED STACK tcpjobname - count MESSAGES WRITTEN TO JOBLOG

Explanation
When running on an MLACTIVE system, TCPIP performs a multilevel security consistency check in any of the following situations:

- After initial profile processing.
- After every VARY TCPIP,,OBEYFILE command.
- After SYSPLEX dynamic VIPA changes.
- When it receives an ENF signal from RACF that indicates that a RAclist REFRESH was done for the SERVAUTH or SECLABEL class.

At least one configuration inconsistency was found and reported in a message.

tcpjobname is the TCPIP JOBNAME.

count is the number of problem messages written to the job log.

System action
The GLOBALCONFIG [NO]MLSCHKTERMINATE setting in TCPIP PROFILE determines the action, as follows:

- MLSCHKTERMINATE indicates that this stack will end because of the reported errors.
- NOMLSCHKTERMINATE indicates that this stack will not end even though it cannot correctly enforce multilevel security policies.

Operator response
Save the job log and contact the system programmer.

System programmer response
Review the MLSCHK messages in the job log. Correct the indicated problems in TCPIP PROFILE or security server profiles. See z/OS Communications Server: IP Diagnosis Guide for more information about diagnosing TCPIP access control problems.

Module
EZBTIMDF

Procedure name
MLSCheck

EZD1218I  MLSCHK ENDED STACK tcpjobname

EZD1xxx messages 825
Explanation
This message is written when a stack that was performing multilevel security consistency checks determines that the system is now NOMLACTIVE. This stack discontinued checking TCP/IP and security server configurations for multilevel security consistency.

tcpjobname is the name of the TCPIP job.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZBTIMDF

Procedure name
MLSCheck

EZD1219I  MLSCHK NETACCESS NOT CONFIGURED WHEN MLACTIVE

Explanation
The TCPIP PROFILE for this stack does not contain a valid NETACCESS statement when running in a multilevel secure environment. The stack does not enforce multilevel security policies when NETACCESS is not configured. The NETACCESS statement must specify both INBound and OUTBound and must contain at least one valid security zone definition.

System action
TCPIP continues.

Operator response
Contact the system programmer.

System programmer response
Add a NETACCESS INBound OUTBound statement with at least one network security zone to the PROFILE for this stack.

Module
EZBTIMDF

Procedure name
MLSCheck

EZD1220I  MLSCHK REQUIRED SERVAUTH PROFILE NOT FOUND RESNM resourcename
Explanation
In a multilevel secure environment, a required security product profile could not be found.

resourcename is the name of the resource that must be covered by a profile in the SERVAUTH class.

System action
TCPIP continues.

Operator response
Contact the system programmer.

System programmer response
Define a security product profile that covers the resource name in the SERVAUTH class. Verify that the SERVAUTH class is active. If changes have been made, issue RACF command SETROPTS RACLIST(SERVAUTH) REFRESH or equivalent security product command. See z/OS Communications Server: IP Configuration Guide for information about the syntax of resourcename.

Module
EZBTIMDF

Procedure name
MLSCheck

EZD1221I  MLSCHK STACKACCESS PROFILE HAS WRONG SECLABEL profsl RESNM resourcename PRFNM profilename

Explanation
While TCPIP is running in a multilevel secure environment, the STACKACCESS profile must have the same security label as the TCPIP job. The TCPIP job security label is displayed in message EZD1215I. The security label must remain defined and active for the duration of the TCPIP job. If any changes are made to either the seclabel class or the servauth class, that class must be refreshed.

profsl is the security label defined in the SECLABEL field in this SERVAUTH profile. If the SECLABEL field is blank, profsl will be <NONE>.

resourcename is the name of the STACKACCESS resource.

profilename is the SERVAUTH profile found for this resource.

System action
TCPIP continues.

Operator response
Contact the system programmer.

System programmer response
Verify that the security label is defined, and active on this system. If necessary, alter the SECLABEL on the profile in the SERVAUTH class to the same security label as the TCPIP job. You might need to define a less generic profile that covers this resource name in the SERVAUTH class. If any changes are made to either the seclabel class or the servauth class, that class must be refreshed. Issue RACF command SETROPTS RACLIST(class name) REFRESH. See z/OS Communications Server: IP Diagnosis Guide for more information about diagnosing TCPIP access control problems.
Module
EZBTIMDF

Procedure name
MLSCheck

**EZD1222I**  MLSCHK LOCAL INTERFACE IS NOT IN A NETWORK SECURITY ZONE IPADR *ipaddress* IF *ifcname*

Explanation
In a multilevel secure environment, all interface addresses must be configured into a NETACCESS security zone. *ipaddress* is the IP address on the local interface. *ifcname* is the name of the INTERFACE or IPv4 LINK statement that defined this IP address.

System action
TCPIP continues.

Operator response
Contact the system programmer.

System programmer response
Verify that this IP address is intended to be defined to this TCPIP stack. If so, modify the NETACCESS statement so that it maps the IP address into the correct security zone. See z/OS Communications Server: IP Configuration Guide for information about configuring Network Access Control.

Module
EZBTIMDF

Procedure name
MLSCheck

**EZD1223I**  MLSCHK LOCAL ZONE HAS INCORRECT SECLABEL *zonesl* IPADR *ipaddress* IF *ifcname* ZONE *zonename* zoneentry RESNM *resourcename* PRFNM *profilename*

Explanation
In a multilevel secure environment, an interface is configured into a NETACCESS security zone with an incorrect security label for the TCPIP job that owns the interface. See message EZD1215I for the security label of the TCPIP job. Local interface addresses must be in zones with the same security label as the TCPIP job. VIPAs on an unrestricted stack may be in a security zone with any security label that is currently active on that system. VIPAs on a restricted stack must be in zones with a security label that is not SYSMULTI and is equivalent to the TCPIP job.

*zonesl* is the security label defined in the SECLABEL field in this SERVAUTH profile. If the SECLABEL field is blank, *zonesl* will be <NONE>.

*ipaddress* is the IPv4 or IPv6 address on the local interface.

*ifcname* is the name of the INTERFACE or IPv4 LINK statement that defined this IP address.

*zonename* is the security zone name defined in the NETACCESS zone entry for this IP address.

*zoneentry* is the NETACCESS zone entry for this IP address. *zoneentry* will one of the following:
- DEFAULT
• DEFAULTHOME
• ipaddress/masklength

resourcename is the name of the SERVAUTH resource.

profilename is the name of the SERVAUTH profile that covers this resource name.

**System action**
TCPIP continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Verify that:
• The IP address is intended to be owned by this TCPIP JOB.
• The NETACCESS statement maps the IP address into the correct security zone.
• The resource name is covered by the intended profile in the SERVAUTH class.
• The profile has the correct security label defined.
• The security label is defined and active on this system.

See z/OS Communications Server: IP Diagnosis Guide for more information about diagnosing TCPIP access control problems.

**Module**
EZBTIMDF

**Procedure name**
MLSCheck

---

**EZD1224I**  MLSCHK STACK SECLABEL stksl IS NOT VALID

**Explanation**
When running on an MLACTIVE system, the TCPIP job must run under a user ID with a security label. The security label must remain defined and active while TCPIP is running.

stksl is the security label of this TCPIP job, or <NONE> if the job was started without a security label.

**System action**
TCPIP continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Verify that:
• The stack is running under the correct user ID.
• The USER profile has the correct default seclabel.
• The security label is defined and active on this system.
• The user ID has READ authority to the security label.
If the user ID or user security label are incorrect, stop TCPIP, correct the problem and start it again.

Module
EZBTIMDF

Procedure name
MLSCheck

**EZD1225I**  MLSCHK *iptype ipaddr* IS NOT IN A NETWORK SECURITY ZONE

**Explanation**

In a multilevel secure environment, *ipaddr* must be configured into a NETACCESS security zone with a security label the same as the TCPIP job.

*iptype* is either INADDR_ANY, IN6ADDR_ANY or TCPSTACKSOURCEVIPA.

*ipaddr* is the associated IP address.

**System action**

TCPIP continues.

**Operator response**

Contact the system programmer.

**System programmer response**

If the *iptype* is TCPSTACKSOURCEVIPA, verify that this IP address is intended to be used by this TCPIP JOB. Modify the NETACCESS statement so that it maps the IP address into the correct security zone.

Module
EZBTIMDF

Procedure name
MLSCheck

**EZD1226I**  MLSCHK *iptype ipaddr* HAS INCORRECT SECLABEL *zonesl* ZONE *zonename* zoneentry RESNM *resourcename* PRFNM *profilenname*

**Explanation**

In a multilevel secure environment, *ipaddr* must be configured into a NETACCESS security zone with a security label the same as the TCPIP job. See message EZD1215I for the security label of the TCPIP job.

*iptype* is one of INADDR_ANY, IN6ADDR_ANY or TCPSTACKSOURCEVIPA.

*ipaddr* is the associated IP address.

*zonesl* is the security label defined in the SECLABEL field in this SERVAUTH profile. If the SECLABEL field is blank, *zonesl* will be <NONE>.

*zonename* is the security zone name defined in the NETACCESS zone entry for this IP address.

*zoneentry* is the NETACCESS zone entry for this IP address. *zoneentry* is either:

- DEFAULT
- DEFAULTHOME
- ipaddress/masklength.
resourcename is the name of the SERVAUTH resource.
profilename is the name of the SERVAUTH profile that covers this resource name.

System action
TCPIP continues.

Operator response
Contact the system programmer.

System programmer response
If the iptype is TCPSTACKSOURCEVIPA, verify that this IP address is intended to be used by this TCPIP job. Verify that:

- The NETACCESS statement maps the IP address into the correct security zone.
- The resource name is covered by the intended profile in the SERVAUTH class.
- The profile has the correct security label defined.
- The security label is defined and active on this system.

See z/OS Communications Server: IP Diagnosis Guide for more information about diagnosing TCPIP access control problems.

Module
EZBTIMDF

Procedure name
MLSCheck

EZD1227I MLSCHK NETACCESS value NOT CONFIGURED WHEN MLACTIVE

Explanation
The TCPIP PROFILE for this stack does not contain a valid NETACCESS statement when running in a multilevel secure environment. The stack does not fully enforce multilevel security policies when NETACCESS is not completely configured. The NETACCESS statement must specify both INBound and OUTBound and must contain at least one valid security zone definition.

value is one of the following:

- INBOUND to indicate that the NETACCESS statement INBound parameter is missing.
- OUTBOUND to indicate that the NETACCESS statement OUTBound parameter is missing.
- SECURITY ZONE to indicate that the NETACCESS statement does not contain any zone entries.

System action
TCPIP continues.

Operator response
Contact the system programmer.

System programmer response
Add a NETACCESS INBound OUTBound statement with at least one network security zone to the PROFILE for this stack.
Module
EZBTIMDF

Procedure name
MLSCheck

**EZD1228I**  MLSCHK INTERFACE ifcname AUTOCONFIGURED WHEN MLACTIVE

**Explanation**
In a multilevel secure environment, an IPv6 interface is found to be incorrectly configured to allow IP addresses to be dynamically autoconfigured. In a multilevel secure network all IP addresses must be manually configured so they can be correctly configured into NETACCESS security zones. All IPv6 INTERFACE statements must specify INTFID to complete the Link Local address. For all IPv6 interface types that support autoconfiguration, the INTERFACE statement must specify at least one IPADDR to disable autoconfiguration. If IPCONFIG6 DYNAMICXCF is configured, it must also specify INTFID to complete the Link Local address.

_ifcname_ is the name of the INTERFACE statement that incorrectly allows autoconfigured addresses.

**System action**
Processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Verify the following:
- The named INTERFACE statement specifies a manual interface ID with the INTFID parameter.
- If IPCONFIG6 DYNAMICXCF is configured, the statement also specifies a manual interface ID with the INTFID parameter.
- If the named INTERFACE statement is of a type that supports dynamic address autoconfiguration, the statement also specifies at least one manually configured prefix or IP address with the IPADDR parameter.

See z/OS Communications Server: IP Configuration Guide for more information about IPv6 dynamic address autoconfiguration.

---

Module
EZBTIMDF

Procedure name
MLSCheck

**EZD1231I**  applname STARTING

**Explanation**
The application is starting.

In the message text:
_applname_
The name of the application. The application name can be one of the following:
- ADNR for the automated domain name registration (ADNR) application
- LBADV for the z/OS Load Balancing Advisor (Advisor)
• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Advisor or Agent, if it is configured for subplexing

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
None.

**Problem determination**
None.

**Source**
z/OS Communication Server TCP/IP other application

**Module**
lamain.c, lmmain.c, ldmain.c

**Routing code**
10

**Descriptor code**
12

**Example**
None.

**EZD1232I applname INITIALIZATION COMPLETE**

**Explanation**
The application completed initialization.

In the message text:

*applname*  
The name of the application. The application name can be one of the following:
• ADNR for the automated domain name registration (ADNR) application
• LBADV for the z/OS Load Balancing Advisor (Advisor)
• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Advisor or Agent, if it is configured for subplexing
System action
Processing continues.

Operator response
None.

System programmer response
None.

User response
None.

Problem determination
None.

Source
z/OS Communication Server TCP/IP other application

Module
lamain.c, lmmain.c, ldmain.c

Routing code
10

Descriptor code
12

Example
None.

EZD1233I applname SHUTDOWN IN PROGRESS

Explanation
The application is shutting down in response to a STOP command.

In the message text:
applname
The name of the application. The application name can be one of the following:
- ADNR for the automated domain name registration (ADNR) application
- LBADV for the z/OS Load Balancing Advisor (Advisor)
- LBAGENT for the z/OS Load Balancing Agent (Agent)
- The job name of the Advisor or Agent, if it is configured for subplexing

System action
Processing continues.
Operator response
None.

System programmer response
None.

User response
None.

Problem determination
None.

Source
z/OS Communication Server TCP/IP other application

Module
lamain.c, lmmain.c, ldmain.c

Routing code
10

Descriptor code
12

Example
None.

**EZD1234I  applname SHUTDOWN COMPLETE**

Explanation
The application ended in response to a STOP command.

In the message text:

*applname*

The name of the application. The application name can be one of the following:

- ADNR for the automated domain name registration (ADNR) application
- LBADV for the z/OS Load Balancing Advisor (Advisor)
- LBAGENT for the z/OS Load Balancing Agent (Agent)
- The job name of the Advisor or Agent, if it is configured for subplexing

System action
The application ends.

Operator response
None.
System programmer response
None.

User response
None.

Problem determination
None.

Source
z/OS Communication Server TCP/IP other application

Module
lamain.c, lmmain.c, ldmain.c

Routing code
10

Descriptor code
12

Example
None.

EZD1235I applname CONFIGURATION ERRORS DETECTED

Explanation
The application configuration file cannot be opened, or contains one or more errors.

In the message text:
applname
The name of the application. The application name can be one of the following:
• ADNR for the automated domain name registration (ADNR) application
• LBADV for the z/OS Load Balancing Advisor (Advisor)
• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Advisor or Agent, if it is configured for subplexing

System action
The application ends.

Operator response
Contact the system programmer.

System programmer response
Examine the syslogd file for the application configuration error messages. The syslogd identifier is lbadv for the Advisor, lbagent for the Agent, and adnr for the automated domain name registration application. Correct the configuration file errors and restart the application. See the information about the Load Balancing Advisor and
User response
Contact the system programmer.

Problem determination
Not applicable.

Source
z/OS Communication Server TCP/IP other application

Module
laconfig.c, lmconfig.c, ldconfig.c

Routing code
10

Descriptor code
12

Example
None.

EZD1236I  applname MODIFY COMMAND ACCEPTED

Explanation
The application accepted a MODIFY command for initial processing.

In the message text:
applname  The name of the application. The application name can be one of the following:
• ADNR for the automated domain name registration (ADNR) application
• LBADV for the z/OS Load Balancing Advisor (Advisor)
• LAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Advisor or Agent, if it is configured for subplexing

System action
Processing continues.

Operator response
None.

System programmer response
None.

User response
None.
Problem determination
None.

Source
z/OS Communication Server TCP/IP other application

Module
lacmd.c, lmcmd.c, ldcmd.c

Routing code
10

Descriptor code
12

Example
None.

EZD1237I  applname MODIFY COMMAND SYNTAX ERROR AT 'location'

Explanation
The application detected a syntax error in a MODIFY command.

In the message text:

applname
The name of the application. The application name can be one of the following:
• ADNR for the automated domain name registration (ADNR) application
• LBADV for the z/OS Load Balancing Advisor (Advisor)
• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Advisor or Agent, if it is configured for subplexing

location
The part of the entered command that is not syntactically valid.

System action
The MODIFY command is rejected.

Operator response
Re-enter the MODIFY command using valid syntax. See the information about the Load Balancing Advisor MODIFY command or the Load Balancing Agent MODIFY command or automated domain name registration Application in z/OS Communications Server: IP System Administrator's Commands for more information.

System programmer response
None.

User response
Re-enter the MODIFY command using valid syntax. See the information about the Load Balancing Advisor MODIFY command or the Load Balancing Agent MODIFY command or automated domain name registration Application in z/OS Communications Server: IP System Administrator’s Commands for more information.
Problem determination
None.

Source
z/OS Communication Server TCP/IP other application

Module
lacmd.c, lccmd.c, lmcmd.c, ldcmd.c

Routing code
10

Descriptor code
12

Example
f adnr,disp,dns,max=1,summary
EZD12371 ADNR MODIFY COMMAND SYNTAX ERROR AT 'SUMMARY'

EZD1238I applname MODIFY COMMAND PARAMETER parm INCORRECT VALUE value

Explanation
The z/OS Load Balancing Advisor (Advisor) or z/OS Load Balancing Agent (Agent) detected an incorrect value for a parameter on a MODIFY command.

In the message text:

applname
  The name of the application. Possible values are:
  • LBADV for the z/OS Load Balancing Advisor (Advisor)
  • LBAGENT for the z/OS Load Balancing Agent (Agent)
  • The job name of the Advisor or Agent, if it is configured for subplexing

parm
  The name of the MODIFY command parameter that is in error.

value
  The incorrect value entered on the MODIFY command.

System action
The MODIFY command is rejected.

Operator response
Re-enter the MODIFY command with a valid value for the indicated parameter. See the information about the Load Balancing Advisor MODIFY command or the Load Balancing Agent MODIFY command in z/OS Communications Server: IP System Administrator’s Commands for more information.

System programmer response
None.
Module
lccmd.c

Procedure name
parse_modify_command

EZD1239I  applname type CALL FAILED errno/errnojr FOR PORT port

Explanation
The z/OS Load Balancing Advisor (Advisor) or z/OS Load Balancing Agent (Agent) was unable to initialize a listening socket.

In the message text:

applname
The name of the application. Possible values are:
• LBADV for the z/OS Load Balancing Advisor (Advisor)
• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Advisor or Agent, if it is configured for subplexing

type
The type of call that failed. For example, the type might be a SOCKET, SETSOCKOPT, or BIND call.

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

errnojr
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

port
The port number on which the call failed.

System action
The application ends.

Operator response
Check to see whether TCP/IP is available. Restart TCP/IP if necessary. Restart the application. If TCP/IP is available when the application issues this message, save the syslogd file and contact the system programmer.

System programmer response
If TCP/IP is available when the application issues this message, examine the syslogd file and correct the error.

Module
lmlisten.c

Procedure name
lmlisten_sock_init

EZD1240I  applname UNABLE TO ESTABLISH endpoint LISTENING SOCKET
Explanation
The z/OS Load Balancing Advisor (Advisor) or z/OS Load Balancing Agent (Agent) was unable to initialize because it could not get a listening socket.

In the message text:

applname
The name of the application. Possible values are:
• LBADV for the z/OS Load Balancing Advisor (Advisor)
• The job name of the Advisor, if it is configured for subplexing

endpoint
The endpoint value can be LOAD BALANCER or AGENT. AGENT is a z/OS Load Balancing Agent.

System action
The application ends.

Operator response
Check to see whether TCP/IP is available. Restart TCP/IP if necessary. Restart the application. If TCP/IP is available when the application issues this message, save the syslogd file and contact the system programmer.

System programmer response
If TCP/IP is available when the application issues this message, examine the syslogd file and correct the error.

Module
lmlisten.c

Procedure name
socket_listener

EZD1241I applname DEBUG LEVEL level

Explanation
This message is issued in response to a z/OS Load Balancing Advisor (Advisor), z/OS Load Balancing Agent (Agent), or z/OS automated domain name registration (ADNR) MODIFY procname,DISPLAY,DEBUG command and the ADNR MODIFY procname,DEBUG,LEVEL command.

If the applname value is the automated domain name registration application, then the enumerated debug levels are also displayed. These enumerated debug settings are displayed on a line separate from the message text.

In the message text:

applname
The name of the application. The application name can be one of the following:
• ADNR for the automated domain name registration (ADNR) application
• LBADV for the z/OS Load Balancing Advisor (Advisor)
• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Advisor or Agent, if it is configured for subplexing

level
The current debug level in effect. Possible values are:

1
Errors are logged.
Warnings are logged.

Significant events are logged.

Informational messages are logged.

Debug data messages are logged. This level is intended for IBM service use only.

- If the applname value is LBADV, then messages related to TCP/IP messages sent between the Advisor and load balancers, between the Advisor and ADNR, and between the Advisor and Agent are logged.
- If the applname value is ADNR, then messages related to the detailed contents of message packets that are sent between the Global Workload Manager (GWM) and the automated domain name registration application are logged.

Debug data messages are logged. This level is intended for IBM service use only.

- If the applname value is LBAGENT, then messages related to data collection and manipulation that support weight calculations are logged.
- If the applname value is ADNR, then messages related to the details of ADNR managing the data in the domain name server zones are logged.

Internal debug data are logged. This level is intended for IBM service use only.

Function entry and exit tracing is logged. This level is intended for IBM service use only.

Individual values can be added together. For example, if the debug level displayed is 7, all error, warning, and event messages are logged. These messages are logged to the syslogd file.

System action
Processing continues.

Operator response
None.

System programmer response
None.

User response
None.

Problem determination
None.

Source
z/OS Communication Server TCP/IP other application

Module
lacmd.c, lmcmd.c, ldcmd.c
Routing code
10

Descriptor code
12

Example

f lbadv,display,debug
EZD1241I LBADV DEBUG LEVEL 7
f adnr,display,debug
EZD1241I ADNR DEBUG LEVEL 15
LOGGING LEVELS : ERROR,WARNING,EVENT,INFO

EZD1242I  LOAD BALANCER SUMMARY

Explaination
The z/OS Load Balancing Advisor (Advisor) issues this message in response to a MODIFY procname,DISPLAY,LB command. This message is followed by summary information for connected load balancers. See the information about the Load Balancing Advisor in z/OS Communications Server: IP System Administrator's Commands for an explanation of the display output.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
lmcmd.c

Procedure name
display_lb_engine

EZD1243I  LOAD BALANCER DETAILS

Explaination
This message is issued in response to a MODIFY procname,DISPLAY,LB,INDEX=index command for the z/OS Load Balancing Advisor (Advisor). The index is a decimal number that identifies the load balancer in console messages. Every load balancer and its index are displayed in response to the Advisor's MODIFY procname,DISPLAY,LB command. This message is followed by detailed information for connected load balancers. See the information about the Load Balancing Advisor in z/OS Communications Server: IP System Administrator's Commands for an explanation of the display output.

System action
Processing continues.
Operator response
None.

System programmer response
None.

Module
lmcmd.c

Procedure name
display_lb_engine

EZD1244I MEMBER SUMMARY

Explanation
This message is issued in response to a MODIFY procname,DISPLAY,MEMBERS command for the z/OS Load Balancing Agent. This message is followed by summary information for members (applications being load balanced) that one or more load balancers registered with the z/OS Load Balancing Advisor.

See the information about the Load Balancing Agent in z/OS Communications Server: IP System Administrator's Commands for an explanation of the display output.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
lacmd.c

Procedure name
display_mem_engine

EZD1245I MEMBER DETAILS

Explanation
This message is issued in response to a MODIFY procname,DISPLAY,MEMBERS,DETAIL command for the z/OS Load Balancing Agent. This message is followed by detailed information for members (applications being load balanced) that one or more load balancers registered with the z/OS Load Balancing Advisor.

See the information about the Load Balancing Agent in z/OS Communications Server: IP System Administrator's Commands for an explanation of the display output.

System action
Processing continues.
Operator response
None.

System programmer response
None.

Module
lacmd.c

Procedure name
display_mem_engine

EZD1246I  applname INITIALIZATION ERROR - REASON CODE reason

Explanation
The application could not initialize because it detected an error.

In the message text:

applname
The name of the application. The application name can be one of the following:
• ADNR for the automated domain name registration (ADNR) application
• LBADV for the z/OS Load Balancing Advisor (Advisor)
• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Advisor or Agent, if it is configured for subplexing

reason
The reason code for the error. Possible values are:

1
Another copy of the application is already active.
• If subplexing is being used:
  – Only one copy of the Advisor can be active in the subplex.
  – On each MVS system, only one copy of the Agent can be active for each subplex on that system.
• If subplexing is not being used:
  – Only one copy of the Advisor can be active in the sysplex.
  – Only one copy of the Agent can be active on each MVS system in the sysplex.
This reason code does not apply to ADNR.

2
The Agent, Advisor, or ADNR application cannot open the configuration file, or found an error in the configuration file. Message EZD1235I should precede this message. See the information about Load Balancing Advisor in z/OS Communications Server: IP Configuration Reference for information about configuring the Advisor, and for information about the Agent. See the information about automated domain name registration in z/OS Communications Server: IP Configuration Reference for information about configuring the ADNR application.

3
Indicates an internal error in the application.

4
The Agent, Advisor, or ADNR application found an error in a start option parameter in the start procedure.
The user ID associated with the started task is not authorized, or the application was not started as a started procedure. See z/OS Communications Server: IP Configuration Guide for information about setting up the authorization profiles for the security product.

A required resource is not available. For example, the TCP/IP stack is not started, or a configured IP address is not defined or available on the TCP/IP stack, or the Advisor or Agent was configured for subplexing and a TCP/IP stack or VTAM node corresponding to the specified TCP/IP sysplex group name could not be found.

System action
The application ends.

Operator response
- If the reason value is 1, then verify that the copy of the application that is active is the one that is wanted. If a previous copy of the application was stopped but never ended, issue a CANCEL command to cancel the previous copy before starting the new copy.
- If the reason value is 6, then start the TCP/IP stack, if it is not already started.
- If the stack is already started, and for all other reason values, save the syslogd file and contact the system programmer.

System programmer response
Take action appropriate for the reason as follows:

1. If subplexing is being used, ensure that the correct TCP/IP sysplex group name is coded in the configuration file for the Agent and Advisor in each subplex.
2. Examine the application syslogd file for errors. Correct the configuration file as needed.
3. Contact IBM software support center. The application syslogd file is the minimum diagnostic data that should be provided. See z/OS Communications Server: IP Diagnosis Guide for information about collecting diagnostic data.
4. Examine the application syslogd file for errors. Correct the start procedure as needed.
5. Examine the security product profiles that have been established for the Advisor, Agent, and ADNR. Ensure that the application was started from a started procedure and that the user ID in the start procedure is permitted to the appropriate profiles if they are defined.
6. Examine the application syslogd file for errors. Verify that the correct IP addresses are configured in the configuration file. Correct the configuration file as needed. See the information about the Load Balancing Advisor in z/OS Communications Server: IP Configuration Reference for more information about configuring the Advisor, and for information about the Agent. See the information about automated domain name registration in z/OS Communications Server: IP Configuration Reference for information about configuring the ADNR application.

User response
- If the reason value is 1, then verify that the copy of the application that is active is the copy that is wanted. If a previous copy of the application was stopped but never ended, issue a CANCEL command to cancel the previous copy before starting the new copy.
- If the reason value is 6, then start the TCP/IP stack, if it is not already started.
• If the stack is already started, and for all other reason values, save the syslogd file and contact the system programmer.

**Problem determination**

To ensure that there is sufficient information to debug this problem, configure the application to use a debug level of 127 and restart the application. See *z/OS Communications Server: IP Configuration Reference* for information about how to configure the debug level for the specified application.

Contact IBM software support center. The application syslogd file is the minimum diagnostic data that should be provided. See *z/OS Communications Server: IP Diagnosis Guide* for information about collecting diagnostic data.

**Problem determination**

Not applicable.

**Source**

z/OS Communication Server TCP/IP other application

**Module**

laimain.c, lmmain.c, ldmain.c

**Routing code**

10

**Descriptor code**

12

**Example**

None.

**EZD1247I applname CONFIGURATION WARNINGS DETECTED**

**Explanation**

The application configuration file contains one or more warnings. These warnings do not cause the application to end, but might indicate conditions that affect the normal operation of the application.

In the message text:

*applname*

The name of the application. The application name can be one of the following:

- ADNR for the automated domain name registration (ADNR) application
- LBADV for the z/OS Load Balancing Advisor (Advisor)
- LBAGENT for the z/OS Load Balancing Agent (Agent)
- The job name of the Advisor or Agent, if it is configured for subplexing

**System action**

Processing continues.

**Operator response**

Save the syslogd file and contact the system programmer.
System programmer response
Examine the syslogd file for the application configuration warning messages. The syslogd identifier is lbadv for the Advisor, lbagent for the Agent, or adnr for the automated domain name registration application. Correct the configuration file warnings. For the Advisor or Agent, restart the application at the earliest opportunity. For the automated domain name registration application, issue the MODIFY REFRESH command. See the information about Load Balancing Advisor and Agent or automated domain name registration in z/OS Communications Server: IP Configuration Reference for more information about configuring the application.

User response
Save the syslogd file and contact the system programmer.

Problem determination
None.

Source
z/OS Communication Server TCP/IP other application

Module
laconfig.c, lmconfig.c, ldconfig.c

Routing code
10

Descriptor code
12

Example
None.

EZD1248I  applname ALL APPLICATIONS ARE ALREADY QUIESCED BY OPERATOR

Explanation
This message is issued in response to a z/OS Load Balancing Agent (Agent) MODIFY procname,QUIESCE or MODIFY procname,ENABLE command. A previous MODIFY procname,QUIESCE,SYSTEM command was issued to quiesce all applications for this Agent.

In the message text:
applname
The name of the application. The application name can be one of the following:
• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Agent, if it is configured for subplexing

System action
The MODIFY command is rejected because all applications are already quiesced.

Operator response
Issue the Agent MODIFY procname,DISPLAY,MEMBERS command to list the registered members on the local MVS system and the status flags for each member. The operator quiesce flag (SYSQ, TCPQ, or APPQ) indicates that an MVS operator quiesced the member. The absence of this flag indicates that the member is enabled from
an MVS operator perspective. To remove the flag, enter a MODIFY, procname, Enable command, and then optionally a MODIFY, procname, QUIESCE command to change the scope of the members quiesced. See the information about the Load Balancing Agent MODIFY command in z/OS Communications Server: IP System Administrator’s Commands for more information.

System programmer response
None.

Module
lacmd.c

Procedure name
eq_engine

EZD1249I  applname ALL APPLICATIONS ARE ALREADY ENABLED BY OPERATOR

Explanation
This message is issued in response to a z/OS Load Balancing Agent (Agent) MODIFY procname, ENABLE, SYSTEM command. A previous MODIFY procname, ENABLE, SYSTEM command was issued to enable all applications for this Agent, or a MODIFY procname, QUIESCE, SYSTEM command was never issued.

In the message text:

applname
The name of the application. The application name can be one of the following:

• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Agent, if it is configured for subplexing

System action
The MODIFY command is rejected because all members in this MVS system are already enabled from an operator perspective. This means that all members are enabled from an MVS operator perspective unless they were individually quiesced by an MVS operator. Members that are individually quiesced by an MVS operator must be individually enabled by the MVS operator.

Operator response
Issue the Agent MODIFY procname, DISPLAY, MEMBERS command to list the registered members on the local MVS system and the status flags for each member. The operator quiesce flag (SYSQ, TCPQ, or APPQ) indicates that an MVS operator quiesced the member. The absence of this flag indicates that the member is enabled from an MVS operator perspective. If the flag value is not wanted, enter a MODIFY procname, QUIESCE or a MODIFY procname, ENABLE command to change the flag value. See the information about the Load Balancing Agent MODIFY command in z/OS Communications Server: IP System Administrator’s Commands for more information.

System programmer response
None.

Module
lacmd.c

Procedure name
eq_engine

EZD1250I  applname ALL APPLICATIONS FOR stackname ARE ALREADY QUIESCED BY OPERATOR
Explanation

This message is issued in response to a z/OS Load Balancing Agent (Agent) MODIFY procname,QUIESCE,TCP=stackname command. A previous MODIFY procname,QUIESCE,TCP=stackname command was issued to quiesce all applications for the TCP/IP stack.

In the message text:

applname

The name of the application. The application name can be one of the following:

- LBAGENT for the z/OS Load Balancing Agent (Agent)
- The job name of the Agent, if it is configured for subplexing

stackname

The name of the TCP/IP stack.

System action

The MODIFY command is rejected because an MVS operator previously quiesced all applications on the stack.

Operator response

Issue the Agent MODIFY procname,DISPAY,MEMBERS command to list the registered members on the local MVS system and the status flags for each member. The operator quiesce flag (SYSQ, TCPQ, or APPQ) indicates that an MVS operator quiesced the member. The absence of this flag indicates that the member is enabled from an MVS operator perspective. If the flag value is not wanted, enter a MODIFY procname,QUIESCE or a MODIFY procname,ENABLE command to change the flag value. See the information about the Load Balancing Agent MODIFY command in z/OS Communications Server: IP System Administrator's Commands for more information.

System programmer response

None.

Module

lacmd.c

Procedure name

eq_engine

EZD1251I  applname ALL APPLICATIONS FOR stackname ARE ALREADY ENABLED BY OPERATOR

Explanation

This message is issued in response to a z/OS Load Balancing Agent (Agent) MODIFY procname,ENABLE,TCP=stackname command. A previous MODIFY procname,ENABLE,TCP=stackname or MODIFY procname,ENABLE,SYSTEM command was issued to enable all applications for the TCP/IP stack or system, or a MODIFY procname,QUIESCE,TCP=stackname command was never issued.

In the message text:

applname

The name of the application. The application name can be one of the following:

- LBAGENT for the z/OS Load Balancing Agent (Agent)
- The job name of the Agent, if it is configured for subplexing

stackname

The name of the TCP/IP stack.
System action
The MODIFY command is rejected because all members for the stack are already enabled from an MVS operator perspective.

Operator response
Issue the Agent MODIFY procname,DISPLAY,MEMBERS command to list the registered members on the local MVS system and the status flags for each member. The operator quiesce flag (SYSQ, TCPQ, or APPQ) indicates that an MVS operator quiesced the member. The absence of this flag indicates that the member is enabled from an MVS operator perspective. If the flag value is not wanted, enter a MODIFY procname,QUIESCE or a MODIFY procname,ENABLE command to change the flag value. See the information about the Load Balancing Agent MODIFY command in z/OS Communications Server: IP System Administrator's Commands for more information.

System programmer response
None.

Module
lacmd.c

Procedure name
eq_engine

EZD1252I applname NO MEMBERS MATCH SELECTION CRITERIA

Explanation
This message is issued in response to a z/OS Load Balancing Agent (Agent) MODIFY procname,QUIESCE or a MODIFY procname,ENABLE command. The MODIFY command is rejected because no active registered target applications match the criteria specified in the MODIFY command.

This message is also issued in response to a MODIFY procname,QUIESCE,TCP=stackname or MODIFY procname,ENABLE,TCP=stackname command when the stackname does not match a TCP/IP stack active on the Agent system.

In the message text:

applname
The name of the application. The application name can be one of the following:

• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Agent, if it is configured for subplexing

System action
The MODIFY command is ignored. Processing continues.

Operator response
Issue the Agent MODIFY procname,DISPLAY,MEMBERS command to list the registered members on the local MVS system and the status flags for each member. The operator quiesce flag (SYSQ, TCPQ, or APPQ) indicates that an MVS operator quiesced the member. The absence of this flag indicates that the member is enabled from an MVS operator perspective. If the flag value is not wanted, enter a MODIFY procname,QUIESCE or a MODIFY procname,ENABLE command to change the flag value. See the information about the Load Balancing Agent MODIFY command in z/OS Communications Server: IP System Administrator's Commands for more information.

System programmer response
None.
Explanation
This message is issued in response to a z/OS Load Balancing Agent (Agent) MODIFY procname,QUIESCE or a MODIFY procname,ENABLE command. The MODIFY command is rejected. A more specific error message precedes this message that explains the reason for the command rejection.

In the message text:

**applname**
The name of the application. Possible values are:
- LBAGENT for the z/OS Load Balancing Agent (Agent)
- The job name of the Agent, if it is configured for subplexing

**command**
The command that was rejected. Possible values are:
- QUIESCE SYSTEM
- ENABLE SYSTEM
- QUIESCE TCPNAME
- ENABLE TCPNAME
- QUIESCE APPLICATION
- ENABLE APPLICATION

System action
The MODIFY command is ignored. Processing continues.

Operator response
Look for a previous message that provides more specific information about why the MODIFY command was rejected. Issue the Agent MODIFY procname,DISPLAY,Members command to list the registered members on the local MVS system and the status flags for each member. The operator quiesce flag (SYSQ, TCPQ, or APPQ) indicates that an MVS operator quiesced the member. The absence of this flag indicates that the member is enabled from an MVS operator perspective. If the flag value is not wanted, enter a MODIFY procname,QUIESCE or a MODIFY procname,ENABLE command to change the flag value. See the information about the Load Balancing Agent MODIFY command in z/OS Communications Server: IP System Administrator's Commands for more information.

System programmer response
None.
EZD1254I  message

Explanation

This message is issued in response to an automated domain name registration (ADNR) MODIFY procname DISPLAY command. This message is followed by information for resources defined to the application.

In the message text:

message

Depending on the command issued, possible values are:

- DNS SUMMARY
- DNS DETAIL
- DNS ZONE SUMMARY
- DNS ZONE DETAIL
- GWM SUMMARY
- GWM DETAIL
- GWM GROUP SUMMARY
- GWM GROUP DETAIL

See the information about automated domain name registration application in z/OS Communications Server: IP System Administrator's Commands for an explanation of the display output.

System action

Processing continues.

Operator response

None.

System programmer response

None.

User response

None.

Problem determination

None.

Source

z/OS Communication Server TCP/IP other application

Module

ldcmd.c

Routing code

10

Descriptor code

12
Example

None.

EZD1255I  applname PORT port IS ALREADY QUIESCED BY OPERATOR

Explanation

This message is issued in response to a z/OS Load Balancing Agent (Agent) MODIFY procname,QUIESCE,PORT=port command. The MODIFY command is rejected because all active registered target applications that match the criteria specified in the MODIFY command are already quiesced.

In the message text:

applname

The name of the application. The application name can be one of the following:

• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Agent, if it is configured for subplexing

port

The local port number that was specified on the MODIFY command.

System action

The MODIFY command is ignored. Processing continues.

Operator response

Issue the Agent MODIFY procname,DISPLAY,MEMBERS command to list the registered members on the local MVS system and the status flags for each member. The operator quiesce flag (SYSQ, TCPQ, or APPQ) indicates that an MVS operator quiesced the member. The absence of this flag indicates that the member is enabled from an MVS operator perspective. If the flag value is not wanted, enter a MODIFY procname,QUIESCE or a MODIFY procname,ENABLE command to change the flag value. See the information about the Load Balancing Agent MODIFY command in z/OS Communications Server: IP System Administrator’s Commands for more information.

System programmer response

None.

Module

lacmd.c

Procedure name

eq_engine

EZD1256I  applname PORT port IS ALREADY ENABLED BY OPERATOR

Explanation

This message is issued in response to a z/OS Load Balancing Agent (Agent) MODIFY procname,ENABLE,PORT=port command. The MODIFY command is rejected because all active registered target applications that match the criteria specified in the MODIFY command are already enabled from an MVS operator perspective.

In the message text:

applname

The name of the application. The application name can be one of the following:

• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Agent, if it is configured for subplexing
**port**
The local port number that was specified on the MODIFY command.

**System action**
The MODIFY command is ignored. Processing continues.

**Operator response**
Issue the Agent MODIFY `procname,DISPLAY,MEMBERS` command to list the registered members on the local MVS system and the status flags for each member. The operator quiesce flag (SYSQ, TCPQ, or APPQ) indicates that an MVS operator quiesced the member. The absence of this flag indicates that the member is enabled from an MVS operator perspective. If the flag value is not wanted, enter a MODIFY `procname,QUIESCE` or a MODIFY `procname,ENABLE` command to change the flag value. See the information about the Load Balancing Agent MODIFY command in z/OS Communications Server: IP System Administrator's Commands for more information.

**System programmer response**
None.

**Module**
lacmd.c

**Procedure name**
eq_engine

**EZD1257I  applname ZONE zoneLabel IS NOT RESPONSIVE**

**Explanation**
The application is not receiving responses from the domain name server zone.

In the message text:

- **applname**
The name of the application. The application name is ADNR for the automated domain name registration (ADNR) application.

- **zoneLabel**
The label of the zone in the application configuration file that is not responding. The zone identified by the `zoneLabel` value occurs in the ADNR configuration, but for reasons involving name server availability, network connectivity, or configuration mismatch, the zone cannot be updated or managed by ADNR

**System action**
ADNR periodically probes this zone to determine whether it is responding. This zone is not updated in the name server until the underlying cause of its unresponsiveness is corrected. ADNR continues processing.

**Operator response**
Contact the system programmer.

**System programmer response**
See the information about diagnosing unresponsive zones in z/OS Communications Server: IP Diagnosis Guide for information about resolving this problem.

**User response**
Not applicable.
Problem determination
Not applicable.

Source
z/OS Communication Server TCP/IP other application

Module
ldzone.c

Routing code
10

Descriptor code
12

Example
None.

EZD1258I  applname AGENT CONNECTION FROM ipaddress CLOSED DUE TO INACTIVITY

Explanation
This immediate action message is issued when the z/OS Load Balancing Advisor (Advisor) closes its connection to the z/OS Load Balancing Agent (Agent) because the Advisor did not receive a message from the Agent in the expected time. The Advisor will delete this message when the Agent connects to the Advisor.

In the message text:

applname
   The name of the application. The application name can be one of the following:
   • LBADV for the z/OS Load Balancing Advisor (Advisor)
   • The job name of the Advisor, if it is configured for subplexing

ipaddress
   The IP address of the Agent.

System action
The Agent will attempt to reconnect to the Advisor until the Agent connects with the Advisor or the Agent is stopped. If the situation is temporary, the Agent will successfully reconnect to the Advisor.

Operator response
If the Agent is not active, start it. If the error persists, contact the system programmer.

System programmer response
Network connectivity problems, routing problems, slow system performance on the Agent system, and low MVS dispatching priority for the Agent might cause this problem. If the error persists, increase the update interval in the Advisor configuration file and restart the Advisor.

Module
lmagnt.c
**EZD1259I**  **applname** CONNECTED TO ADVISOR **ipaddress**

**Explanation**
This message is issued when the z/OS Load Balancing Agent (Agent) is successfully connected to the z/OS Load Balancing Advisor (Advisor).

In the message text:
- **applname**
  The name of the application. The application name can be one of the following:
  - LBAGENT for the z/OS Load Balancing Agent (Agent)
  - The job name of the Agent, if it is configured for subplexing
- **ipaddress**
  The IP address of the Advisor.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
laadv.c

---

**EZD1260I**  **applname** CONNECTION TO ADVISOR **ipaddress** IS NO LONGER ACTIVE

**Explanation**
This message is issued when the z/OS Load Balancing Agent (Agent) loses its connection to the z/OS Load Balancing Advisor (Advisor).

In the message text:
- **applname**
  The name of the application. The application name can be one of the following:
  - LBAGENT for the z/OS Load Balancing Agent (Agent)
  - The job name of the Agent, if it is configured for subplexing
- **ipaddress**
  The IP address of the Advisor.

**System action**
The Agent will attempt to reconnect to the Advisor until the Agent connects with the Advisor or the Agent is stopped. If the situation is temporary, the Agent will successfully reconnect to the Advisor.
Operator response
If the Advisor is not active, start it. If the Advisor is active, save the syslogd file and contact the system programmer.

System programmer response
Examine the syslogd file for Advisor or Agent error or warning messages. The syslogd identifier is lbadv for the Advisor and lbagent for the Agent. Correct any errors and restart the application.

Module
lamain.c

Procedure name
reset_connection

EZD1261I  applname AGENT CONNECTED FROM ipaddress

Explanation
This message is issued when the z/OS Load Balancing Advisor (Advisor) detects that it successfully connected with a z/OS Load Balancing Agent (Agent).

In the message text:

applname
The name of the application. The application name can be one of the following:
• LBADV for the z/OS Load Balancing Advisor (Advisor)
• The job name of the Advisor, if it is configured for subplexing

ipaddress
The IP address of the Agent.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
Imagnt.c

Procedure name
Im_process_agentRegistrationRequest

EZD1262I  applname AGENT CONNECTION FROM ipaddress IS NO LONGER ACTIVE

Explanation
This message is issued when the z/OS Load Balancing Advisor (Advisor) detects that it lost its connection with a z/OS Load Balancing Agent (Agent). This usually indicates that the agent has been stopped.

In the message text:
**applname**
The name of the application. The application name can be one of the following:
- LBADV for the z/OS Load Balancing Advisor (Advisor)
- The job name of the Advisor, if it is configured for subplexing

**ipaddress**
The IP address of the Agent.

**System action**
Processing continues. If the Agent was not stopped, the Agent will attempt to reconnect to the Advisor. If the situation is temporary, the Agent will successfully reconnect to the Advisor.

**Operator response**
Check to see why the Agent has ended. Restart the Agent, if necessary. If the Agent cannot reconnect to the Advisor, save the syslogd file and contact the system programmer.

**System programmer response**
No action is needed if the Agent was stopped. Otherwise, examine the syslogd file for Advisor or Agent error or warning messages. The syslogd identifier is lbadv for the Advisor and lbagent for the Agent. Correct any errors and restart the application.

**Module**
lmagnt.c

**Procedure name**
lm_service_agent_connection

---

**EZD1263I  applname LOAD BALANCER CONNECTED FROM ipaddress**

**Explanation**
This message is issued when the z/OS Load Balancing Advisor (Advisor) detects that it successfully connected with a load balancer.

In the message text:

**applname**
The name of the application. The application name can be one of the following:
- LBADV for the z/OS Load Balancing Advisor (Advisor)
- The job name of the Advisor, if it is configured for subplexing

**ipaddress**
The IP address of the connected load balancer.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
Module
lmlb.c

Procedure name
lm_process_registrationRequest, lm_process_setLbStateRequest

EZD1264I  applname LOAD BALANCER CONNECTION FROM ipaddress IS NO LONGER ACTIVE

Explanation
This message is issued when the z/OS Load Balancing Advisor (Advisor) detects that it lost its connection with a load balancer.

In the message text:

applname
The name of the application. The application name can be one of the following:

• LBADV for the z/OS Load Balancing Advisor (Advisor)
• The job name of the Advisor, if it is configured for subplexing

ipaddress
The IP address of the previously connected load balancer.

System action
If the load balancer is still active, the load balancer might attempt to reconnect to the Advisor.

Operator response
Check to see whether the load balancer is active. Restart the load balancer, if necessary. If the load balancer is active, contact the system programmer.

System programmer response
Check the load balancer and correct any configuration or connectivity problems. See z/OS Communications Server: IP Diagnosis Guide for information about diagnosing network connectivity problems.

Module
lmlb.c

Procedure name
lm_service_lb_connection

EZD1265I  applname CONNECTION TO ADVISOR ipaddress CLOSED DUE TO PROTOCOL ERROR

Explanation
This immediate action message is issued when the z/OS Load Balancing Agent (Agent) closes its connection to the z/OS Load Balancing Advisor (Advisor) due to an internal protocol error. The Agent will delete this message when the Agent connects to the Advisor.

In the message text:

applname
The name of the application. The application name can be one of the following:

• LBAGENT for the z/OS Load Balancing Agent (Agent)
• The job name of the Agent, if it is configured for subplexing
**ipaddress**
The IP address of the Advisor.

**System action**
The Agent will take a CEEDUMP, write messages to the syslogd file, and attempt to reconnect to the Advisor. The Agent will attempt to reconnect to the Advisor until the Agent connects with the Advisor or the Agent is stopped.

**Operator response**
Save the dump, syslogd file, and packet trace (if active). The dump data set is specified in the CEEDUMP DD statement in the Agent's start procedure. Contact the system programmer. If the problem persists, stop the Agent.

**System programmer response**
Examine the syslogd files for the Advisor and Agent.

If the debug level includes Message level messages, the syslogd file will show the data that each application sent or received. If the data that was sent by one application was the same data received by the corresponding application, contact IBM software support services.

If the data that was sent was not the same data that was received, this might indicate that network integrity has been compromised.

If the packet trace was active, examine the trace for the data that was sent and received.

If the packet trace is not active, try to recreate the problem with packet trace active. See *z/OS Communications Server: IP Diagnosis Guide* for information about the packet trace.

**Module**
lczap.c

**Procedure name**
zap_protocol_error

**EZD1266I  applname AGENT CONNECTION FROM ipaddress CLOSED DUE TO PROTOCOL ERROR**

**Explanation**
This immediate action message is issued when the z/OS Load Balancing Advisor (Advisor) closes its connection with a z/OS Load Balancing Agent (Agent) as a result of an internal protocol error. The Advisor will delete this message when the Agent connects to the Advisor.

In the message text:

**applname**
The name of the application. The application name can be one of the following:

- LBADV for the z/OS Load Balancing Advisor (Advisor)
- The job name of the Advisor, if it is configured for subplexing

**ipaddress**
The IP address of the Agent.

**System action**
The Advisor will take a CEEDUMP and write messages to the syslogd file. If the Agent is still active, it will attempt to reconnect to the Advisor until the Agent connects with the Advisor or the Agent is stopped. If the situation is temporary, the Agent will successfully reconnect to the Advisor.
**Operator response**
Save the dump, syslogd file, and packet trace (if active). The dump data set is specified in the CEEDUMP DD statement in the Advisor's start procedure. Contact the system programmer. If the Agent is not active, start it.

**System programmer response**
Examine the syslogd files for the Advisor and Agent. If the debug level includes Message level messages, the syslogd file will show the data that each application sent or received.

If an Agent is configured to use AT-TLS and the Advisor is not, the configuration is not valid. The Advisor will issue message EZD1266I. To correct this configuration, configure the Advisor to use AT-TLS.

If the data that was sent by one application was the same data received by the corresponding application, contact IBM software support services.

If the data that was sent was not the same data that was received, this might indicate that network integrity has been compromised.

If the packet trace was active, examine the trace for the data that was sent and received.

If the packet trace is not active, try to recreate the problem with packet trace active. See z/OS Communications Server: IP Diagnosis Guide for information about the packet trace.

**Module**
lczap.c

**Procedure name**
zap_protocol_error

**EZD1267I applname ENDED ABNORMALLY**

**Explanation**
The application ended in response to an unexpected error.

In the message text:

**applname**
The name of the application. The application name can be one of the following:

- ADNR for the automated domain name registration (ADNR) application
- LBADV for the z/OS Load Balancing Advisor (Advisor)
- LBAGENT for the z/OS Load Balancing Agent (Agent)
- The job name of the Advisor or Agent, if it is configured for subplexing.

**System action**
Processing continues.

**Operator response**
Save the CEEDUMP, snap output, and syslogd file. Contact the system programmer.

**System programmer response**
Contact IBM software support services.

**Module**
lcassert.c, lceror.c

862 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**EZD1268I**  
*applname ZONE zoneLabel IS NOW RESPONDING*

**Explanation**

The domain name server zone that was previously not responding (as reported by message EZD1257I) is now responding to requests from the application.

In the message text:

*applname*

The name of the application. The application name is ADNR for the automated domain name registration (ADNR) application.

*zoneLabel*

The label of the zone in the application configuration file.

**System action**

The application continues processing.

**Operator response**

None.

**System programmer response**

None.

**User response**

None.

**Problem determination**

None.

**Source**

z/OS Communication Server TCP/IP other application

**Module**

ldzone.c

**Routing code**

10

**Descriptor code**

12

**Example**

None.

---

**EZD1269I**  
*applname COMMAND REJECTED - REFRESH IN PROGRESS*
Explanation
The specified application is currently processing a previous MODIFY REFRESH command.

In the message text:

applname
The name of the application. The application name is ADNR for the automated domain name registration (ADNR) application.

System action
The application continues processing.

Operator response
If the application is ADNR, then reissue the command after the active refresh command completes. Message EZD1275I is issued when the refresh is complete. Message EZD1277I is issued if the refresh completes with a failure. If the pending ADNR refresh does not complete, contact the system programmer.

System programmer response
Contact IBM software support services.

User response
Not applicable.

Problem determination
If the application is ADNR and the pending refresh does not complete, then see the information about diagnosing unresponsive zones in z/OS Communications Server: IP Diagnosis Guide for information about why the ADNR refresh failed to complete.

Source
z/OS Communication Server TCP/IP other application

Module
ldcmd.c

Routing code
10

Descriptor code
12

Example
None.

EZD1270I ADNR CONNECTED TO GWM gwmipaddress

Explanation
The automated domain name registration (ADNR) application established a connection to the Global Workload Manager (GWM).

In the message text:
**gwmipaddress**
The IP address of the GWM. The IP address is specified on the gwm_id parameter of the gwm statement in the ADNR configuration file.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP other application

**Module**
ldgwm.c

**Routing code**
10

**Descriptor code**
12

**Example**

```
EZD1270I  ADNR CONNECTED TO GWM 10.42.11.182
```

**EZD1271E  ADNR CONNECTION TO GWM gwmipaddress IS NO LONGER ACTIVE**

**Explanation**
The automated domain name registration (ADNR) application no longer has a connection to the Global Workload Manager (GWM).

In the message text:

- **gwmipaddress**
  The IP address of the GWM. The IP address is specified on the gwm_id parameter of the gwm statement in the ADNR configuration file.

**System action**
Processing continues. The ADNR application can no longer communicate with the GWM. ADNR will periodically try the connection again. If it succeeds, the eventual action message will be deleted. See the information about
**Operator response**

If the GWM application has been stopped, no further actions are needed. When the GWM application is restarted, the ADNR application automatically reconnects to the GWM. If the GWM was not stopped and the reason for the connection attempt failure between the ADNR and GWM application cannot be determined, save the following information:

- The ADNR log file
- Any ADNR dump produced when the problem occurred
- The TCP/IP profile and system log of the TCP/IP stack used by the ADNR application
- The GWM log file
- The TCP/IP profile and system log of the TCP/IP stack used by the GWM application

Contact the system programmer.

**System programmer response**

Verify that routing exists between the IP address used by the ADNR application and the IP address of the GWM that is specified in the message. A route should exist from the ADNR IP address to the GWM IP address, and a route should also exist from the GWM IP address to the ADNR IP address. The IP address used by the ADNR application is found in one of the following places:

- The IP address is specified on the host_connection_addr parameter of the gwm statement that contains the corresponding gwm_id parameter in the ADNR configuration file.
- The IP address is displayed in message EZD1263I. This message is issued by the Load Balancing Advisor. Look for this message in the system log for the system on which the Advisor is running, at a time that corresponds approximately to the time at which this ADNR was started.

If routing exists, then check the ADNR and GWM log files for messages corresponding to the time when the connection failed. See the information about diagnosing problems with the automated domain name registration application in z/OS Communications Server: IP Diagnosis Guide. If the reason for the connection loss cannot be determined, contact the IBM support center after obtaining the supporting TCP/IP, ADNR, and GWM documentation.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communication Server TCP/IP other application

**Module**

ldgwm.c

**Routing code**

1

**Descriptor code**

2
Example

EZD1271E ADNR CONNECTION TO GWM 10.42.11.182 IS NO LONGER ACTIVE

EZD1272E ADNR CONNECTION ATTEMPT TO GWM gwmipaddress FAILED

Explanation
The automated domain name registration (ADNR) application failed to establish a connection to the Global Workload Manager (GWM).

In the message text:

**gwmipaddress**
The IP address of the GWM. The IP address is specified on the gwm_id parameter of the gwm statement in the ADNR configuration file.

System action
Processing continues. The ADNR application cannot communicate with the GWM. ADNR will periodically try the connection again. If it succeeds (as indicated by message EZD1270I), the eventual action message will be deleted. See the information about automated domain name registration in z/OS Communications Server: IP Configuration Guide or diagnosing problems with the automated domain name registration application in z/OS Communications Server: IP Diagnosis Guide for more information.

Operator response
If the GWM application has been stopped, no further actions are needed. When the GWM application is restarted, the ADNR application automatically reconnects to the GWM. If the GWM was not stopped and the reason for the connection attempt failure between the ADNR and GWM application cannot be determined, save the following information:

- The ADNR log file
- Any ADNR dump produced when the problem occurred
- The TCP/IP profile and system log of the TCP/IP stack used by the ADNR application
- The GWM log file
- The TCP/IP profile and system log of the TCP/IP stack used by the GWM application.

Contact the system programmer.

System programmer response
Verify that routing exists between the IP address used by the ADNR application and the IP address of the GWM. A route should exist from the ADNR IP address to the GWM IP address, and a route should also exist from the GWM IP address to the ADNR IP address. The IP address of the GWM is specified in the message. The IP address used by the ADNR application is one of the following:

- Specified on the host_connection_addr parameter of the gwm statement containing the corresponding gwm_id parameter in the ADNR configuration file.
- Selected by the ADNR's TCP/IP stack. If host_connection_addr is not specified, the ADNR attempts to connect to the IP address specified in the gwm_id parameter. The ADNR's TCP/IP stack selects a source IP address. See the information about source IP address selection in z/OS Communications Server: IP Configuration Guide for information about how TCP/IP selects a source IP address.

If routing exists, then check the ADNR and GWM log files for messages corresponding to the time when the connection failed. See the information about diagnosing problems with the automated domain name registration application in z/OS Communications Server: IP Diagnosis Guide. If the reason for the connection attempt failure cannot be determined, contact IBM software support services after obtaining the supporting TCP/IP, ADNR and GWM documentation.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communication Server TCP/IP other application

Module
ldgwm.c

Routing code
1

Descriptor code
2

Example

| EZD1272E  | ADNR CONNECTION ATTEMPT TO GWM 10.42.11.182 FAILED |
---|---|

EZD1273I  ADNR FAILED TO DELETE ZONE zoneLabel

Explanation
The automated domain name registration (ADNR) application failed to delete a domain name server zone that is not in the new configuration while processing a MODIFY procname,REFRESH command.

In the message text:
zoneLabel
The label of the zone in the ADNR configuration file.

System action
Any domain name server resource records from the old configuration remain in the domain name server zone identified by the zoneLabel value. Processing continues. See the information about automated domain name registration in z/OS Communications Server: IP Configuration Guide or diagnosing problems with the automated domain name registration application in z/OS Communications Server: IP Diagnosis Guide for more information.

Operator response
Contact the system programmer.

System programmer response
Perform one of the following actions:
- Use the Dig or Nslookup utility to perform a zone transfer of the zone to examine its contents. Based on the results from using the Dig or Nslookup utility, use the Nsupdate utility to manually remove the resource records. See the information about querying and administrating a Domain Name System in z/OS Communications Server: IP System Administrator's Commands for information about using the Nslookup, Dig, and Nsupdate commands.
• If you can wait for communications to be restored between the ADNR application and the zone's name server, see the information about flushing a zone in z/OS Communications Server: IP Configuration Guide. Before this communication is restored, the name server might contain stale data for this zone.

**User response**
None.

**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP other application

**Module**
ldzone.c

**Routing code**
10

**Descriptor code**
12

**Example**
None.

**EZD1274I  applname NEW CONFIGURATION INFORMATION TEMPORARILY UNAVAILABLE**

**Explanation**
This message is issued in response to the specified application MODIFY DISPLAY command. A dynamic reconfiguration operation (REFRESH) is still in progress. The old configuration is being displayed, and the new (refreshed) configuration is not yet available for display.

In the message text:

**applname**
The name of the application. The application name is ADNR for the automated domain name registration (ADNR) application.

**System action**
The application continues.

**Operator response**
If the application is ADNR, then reissue the ADNR MODIFY DISPLAY command after the ADNR MODIFY REFRESH command has completed (as indicated when message EZD1275I ADNR REFRESH COMMAND COMPLETED is issued).

**System programmer response**
None.
**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP other application

**Module**
ldcmd.c

**Routing code**
10

**Descriptor code**
12

**Example**
None.

**EZD1275I applname REFRESH COMMAND COMPLETED**

**Explanation**
The application's MODIFY REFRESH command has completed. Subsequent MODIFY REFRESH commands will now be accepted.

In the message text:

applname
The name of the application. The application name is ADNR for the automated domain name registration (ADNR) application.

**System action**
The new configuration is now active.

**Operator response**
None.

**System programmer response**
None.

**User response**
None.

**Problem determination**
Not applicable.
EZD1276I  ADNR CONNECTION TO GWM gwmipaddress CLOSED DUE TO PROTOCOL ERROR

Explanation

The automated domain name registration (ADNR) application closed the connection to the Global Workload Manager (GWM) as a result of a Server/Application State Protocol (SASP) protocol violation or a configuration mismatch. One common configuration mismatch is failing to include the IP address used by ADNR in the Advisor lb_id_list. If the lb_id_list statement is not specified in the Advisor's configuration file, AT-TLS must be configured for the connection to be fully established. See the information about automated domain name registration in z/OS Communications Server: IP Configuration Reference for more information about how to correctly coordinate the configuration of ADNR and the GWM.

The automated domain name registration (ADNR) application closed the connection to the Global Workload Manager (GWM) as a result of a Server/Application State Protocol (SASP) protocol violation.

In the message text:

\texttt{gwmipaddress}

The IP address of the GWM. The IP address is specified on the gwm_id parameter of the gwm statement in the ADNR configuration file.

System action

Processing continues. ADNR dumps its address space. The ADNR application is no longer able to communicate with the GWM. ADNR will periodically reestablish the connection to the GWM and attempt to communicate with the GWM. See the information about automated domain name registration in z/OS Communications Server: IP Configuration Guide and diagnosing problems with the automated domain name registration application (EZBADNR) in the z/OS Communications Server: IP Diagnosis Guide for more information.

Operator response

Save the ADNR log file, and the ADNR dump. Save the GWM log file. Contact the system programmer.

System programmer response

Review the ADNR log file and the GWM log file to determine the cause for the connection being closed. Correct any configuration mismatch errors. If the error is a SASP protocol violation, contact IBM software support services after obtaining the supporting ADNR and GWM documentation.

User response

Not applicable.
**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP other application

**Module**
ldgwm.c

**Routing code**
10

**Descriptor code**
12

**Example**
```
EZD1276E ADNR CONNECTION TO GWM 10.42.11.182 CLOSED DUE TO PROTOCOL ERROR
```

**EZD1277I  applname REFRESH COMMAND FAILED**

**Explanation**
The application MODIFY REFRESH command failed because of errors in the configuration file. The new configuration is not used. Subsequent MODIFY REFRESH commands can now be accepted.

In the message text:

`applname`  
The name of the application. The application name is ADNR for the automated domain name registration (ADNR) application.

**System action**
Processing continues. The previous configuration continues to be used.

**Operator response**
Contact the system programmer.

**System programmer response**
Check the application log files for error messages to determine why the configuration failed.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP other application
Module
ldconfig.c

Routing code
10

Descriptor code
12

Example
None.

**EZD1278E**  *applname A ZONE SUBORDINATE TO DNS dnsLabel IS NOT RESPONSIVE*

**Explanation**
One or more domain name server zones are not responding to the specified application.

In the message text:

**applname**
- The name of the application. The application name is ADNR for the automated domain name registration (ADNR) application.

**dnsLabel**
- Label of the domain name server server in the application configuration file containing one or more unresponsive zones.

**System action**
The application continues.

**Operator response**
If the application is ADNR, then review the ADNR system log for the reason why the zone is not responsive to the ADNR application. An EZD1257I message is issued to identify each zone that is not responsive.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communication Server TCP/IP other application

Module
lddns.c
Routing code
1

Descriptor code
2

Example
None.

EZD1279I  applname  SECURE CONNECTION REQUEST RECEIVED FROM USER  user_id  AT IP ADDRESS  ip_addr

Explanation
The Advisor received a Transport Layer Security (TLS) secure connection request at the specified IP address from the user specified by the  user_id  value. SAF authorization for access to the Advisor will be checked for the specified user ID.

In the message text:

applname
   The name of the application that received the connection request. Possible values are:
      • LBADV for the z/OS Load Balancing Advisor (Advisor)
      • The job name of the Advisor, if it is configured for subplexing

user_id
   The user identifier of the load balancer or Load Balancing Agent requesting access to the Advisor.

ip_addr
   The IP address of the load balancer or Load Balancing Agent.

System action
The system continues processing.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Load Balancing Advisor

Module
Immain
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
LBADV SECURE CONNECTION REQUEST RECEIVED FROM USER AGENT1 AT IP ADDRESS 192.10.1.1

EZD1280I applname client CONNECTION ATTEMPT FROM USER userid AT IP ADDRESS ip_addr FAILED REASON CODE reason

Explanation
The Advisor received a connection request at the specified IP address at the specified IP address from an Agent or load balancer with the specified user ID. Authorization for connection to the Advisor failed for the specified user ID.

In the message text:

applname
The name of the application that received the connection request. Possible values are:
• LBADV for the z/OS Load Balancing Advisor (Advisor).
• The job name of the Advisor, if it is configured for subplexing.

client
The type of client that attempted to connect to the Advisor. Possible values are:
• AGENT for the z/OS Load Balancing Agent
• LB for a load balancer or ADNR connection

userid
The user ID of the load balancer or Load Balancing Agent that is requesting access to the Advisor. If the user ID is not obtained from AT-TLS, the value is UNKNOWN.

ip_addr
The IP address of the load balancer or Load Balancing Agent.

reason
A code that explains the failure. Possible values are:

1
The Advisor TCP/IP stack is not configured for Application Transparent Transport Layer Security (AT-TLS), and the Advisor configuration file did not allow connections from this client. The TTLS option in the TCP/IP profile TCPCONFIG statement enables the stack for AT-TLS.

2
There is not a usable AT-TLS policy for this connection, and the Advisor configuration file did not allow connections from this client. For example, the policy agent is not active, or the AT-TLS policy for this connection specifies the wrong port.

3
The AT-TLS policy defined for this connection does not enable AT-TLS, and the Advisor configuration file did not allow connections from this client. In the policy, the TLSSGroupAction statement is not configured with TTLSEnabled set to ON.
The AT-TLS policy that is defined for this connection does not define the Advisor as a controlling application, and the Advisor configuration file did not allow connections from this client. In the policy, the TTLSEnvironmentAdvancedParms parameter is not configured with ApplicationControlled set to On for the Advisor.

The AT-TLS handshake failed for this connection, and the Advisor configuration file did not allow connections from this client.

System authorization facility (SAF) authorization failed for this connection. The SERVAUTH class profile EZB.LBA.LBACCESS.sysname.tcpysplexgroupname (for a load balancer connection) or EZB.LBA.AGENTACCESS.sysname.tcpysplexgroupname (for an Agent connection) exists but the user is not authorized to use this profile. The system does not use the Advisor configuration file because the user is not authorized to use the SERVAUTH class profile.

The Advisor was unable to obtain storage for processing an AT-TLS connection request, and the Advisor configuration file did not allow connections from this client.

The Advisor call to the SIOCTTLSCTL IOCTL failed unexpectedly, and the Advisor configuration file did not allow connections from this client.

System authorization facility (SAF) authorization failed for this connection, and the Advisor configuration file did not allow connections from this client. The SERVAUTH class profile EZB.LBA.LBACCESS.sysname.tcpysplexgroupname (for a load balancer connection) or EZB.LBA.AGENTACCESS.sysname.tcpysplexgroupname (for an Agent connection) is not protected by SAF.

System action
The system continues processing. The client that attempted to connect to the Advisor might continue to attempt to connect.

Operator response
If you are not using AT-TLS for this connection, save the Advisor syslogd file and contact the system programmer. If you are using AT-TLS for this connection, take the action appropriate for the reason as follows:

<table>
<thead>
<tr>
<th>reason</th>
<th>action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Start the Policy Agent if it is not already started. If the AT-TLS policy for the Advisor connections has changed, refresh the Policy Agent. If the problem is not corrected, save the Advisor syslogd file, the AT-TLS syslogd file, and the policy agent syslogd file, then contact the system programmer.</td>
</tr>
<tr>
<td>7</td>
<td>If the storage problem cannot be corrected, save the Advisor syslogd file. If a dump was not created, take a dump of the Advisor address space, then contact the system programmer.</td>
</tr>
</tbody>
</table>

All other reasons
Save the system console, the Advisor syslogd file, the AT-TLS syslogd file, and the policy agent syslogd file, then contact the system programmer.

See z/OS Communications Server: IP Diagnosis Guide for information about collecting diagnostic data.
System programmer response

If you are not using AT-TLS, examine the Advisor syslogd file for errors. Correct the configuration file as needed. See z/OS Communications Server: IP Configuration Reference for information about configuring the Advisor and Agent and ADNR application.

If you are using AT-TLS for this connection, take action appropriate for the reason as follows:

1. Activate AT-TLS with the TCPCONFIG TTLS configuration statement. Either correct and resubmit the original TCP/IP profile or submit a VARY TCPIP, OBEYFILE command. See the information about the TCPCONFIG statement in z/OS Communications Server: IP Configuration Reference for more information about the TTLS parameter.

2. If the Policy Agent is active and has been refreshed since the last change to the AT-TLS policy, examine the system console, the Advisor syslogd file, the AT-TLS syslogd file, and the policy agent syslogd file for errors. Correct the AT-TLS policy for this connection. See the information about diagnosing AT-TLS problems in z/OS Communications Server: IP Diagnosis Guide and Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Refresh the Policy Agent after changing the policy.

3. Change the AT-TLS policy for this connection in the TTLSEnableGroupAction statement to TTLSEnabled On. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Refresh the Policy Agent after changing the policy.

4. Change the AT-TLS policy for this connection in the TTLSEnvironmentAdvancedParms statement to ApplicationControlled On for the server (Advisor). See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Refresh the Policy Agent after changing the policy.

5. Correct the TLS handshake parameters in the AT-TLS policy for this connection.
   • See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference. Refresh the Policy Agent after changing the policy. For example,
     – Ensure that the HandshakeTimeout value for the Advisor policy is sufficient (for example, 30 seconds)
     – Ensure that the HandshakeRole value for the Advisor is ServerWithClientAuth or Server.
     – Ensure that the HandshakeRole value for the Agent and load balancers is Client.

6. Ensure that the user ID has at least read access to the correct SERVAUTH class profile (EZB.LBA.LBACCESS.sysname.tcpsysplexgroupname for a load balancer connection, EZB.LBA.AGENTACCESS.sysname.tcpsysplexgroupname for an Agent connection). For more information, see z/OS Security Server RACF Command Language Reference.

7. If the storage problem cannot be corrected, contact IBM software support services with all supporting documentation. The application syslogd file is the minimum diagnostic data that should be provided. See z/OS Communications Server: IP Diagnosis Guide for information about collecting diagnostic data.

8. Examine the system console, the Advisor syslogd file, the AT-TLS syslogd file, and the policy agent syslogd file for errors. Ensure that the certificate is correct. For more information, see z/OS Security Server RACF Command Language Reference. If the problem is not corrected, contact IBM software support services with all supporting documentation. See z/OS Communications Server: IP Diagnosis Guide for information about collecting diagnostic data.

9. Define and permit the LBACCESS and AGENTACCESS profiles on each system where the Advisor can run. Ensure that the user ID has at least read access to the correct SERVAUTH class profile (EZB.LBA.LBACCESS.sysname.tcpsysplexgroupname for a load balancer connection, EZB.LBA.AGENTACCESS.sysname.tcpsysplexgroupname for an Agent connection). See the z/OS Security
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Load Balancing Advisor

Module
lmmain

Routing code
10

Descriptor code
12

Example
EZD1280I LBADV AGENT CONNECTION ATTEMPT FROM USER AGENT1 AT IP ADDRESS 192.10.1.1 FAILED REASON CODE 6

Explanation
The TCP connection with the specified connection ID (CONNID) matched the specified Application Transparent Transport Layer Security (AT-TLS) rule. This CONNID will be used in all future AT-TLS messages for this connection. The message creation time and owning TCP/IP jobname of the process creating the message are included in the syslog trace prior to the message ID. This message has a syslog priority of INFO and is written to the syslog when AT-TLS trace option INFO (4) or EVENT (8) is specified.

cid is a hexadecimal value that uniquely identifies this TCP connection for the life of the connection.
loc_ip is the local IPv4 or IPv6 address.
loc_port is the local port number.
rem_ip is the remote IPv4 or IPv6 address.
rem_port is the remote port number.
jobname is the job name of the application associated with this connection.
userid is the user ID of the application associated with this connection.
type is the direction of connection initiation for connections using the primary policy mapping method: Inbound if Accepting, Outbound if Connecting. type is Secondary if the SecondaryMap method was used in either direction.
stat is the AT-TLS status for the connection. The values for stat are:
- Not Enabled if TTLSEnabled in the matching AT-TLS policy is set to OFF.
• Enabled if TTLSEnabled in the matching AT-TLS policy is set to ON.
• Appl Control if ApplicationControlled in the matching AT-TLS policy is set to ON.

*rule* is the name of the TTLSRule that mapped this connection.

*grp_act* is the name of the TTLSGroupAction.

*env_act* is the name of the TTLSEnvironmentAction. If a TTLSEnvironmentAction is not specified on the TTLSRule statement, this value will be *N/A*.

*conn_act* is the name of the TTLSConnectionAction. If a TTLSConnectionAction is not specified on the TTLSRule statement, this value will be *N/A*.

**System action**

None.

**Operator response**

None.

**System programmer response**

None.

**Module**

EZBTLMSG

**Procedure name**

EZBTLMSG

**EZD1282I**

| TTLS Start GRPID: gid | ENVID: eid | CONNID: cid | event | ACTIONS: grp_act | env_act | conn_act | add_data |

**Explanation**

An Application Transparent Transport Layer Security (AT-TLS) event has been started for the specified connection ID (CONNID). The message creation time and owning TCP/IP job name of the process creating the message are included in the syslog trace prior to the message ID. This message has a syslog priority of DEBUG and is written to the syslog when AT-TLS trace option EVENT (8) or FLOW (16) is specified. Flow messages (EZD1284I) and Event messages (EZD1283I) might follow this message, depending on the trace options chosen.

*gid* is a hexadecimal value that uniquely identifies the AT-TLS group supporting the connection or SSL environment.

*eid* is a hexadecimal value that uniquely identifies the AT-TLS environment supporting the connection. Multiple AT-TLS environments might be represented by a single master System SSL secure environment. If *eid* is 00000000, the event does not apply to a specific environment.

*cid* is a hexadecimal value that uniquely identifies this TCP connection for the life of the connection. A previously issued message EZD1281I provides additional information about the connection. If *cid* is 00000000, the event does not apply to a specific connection.

*event* is an AT-TLS event. The start of the following events are reported by this message:

**Connection Abend Close**

An abend occurred while executing AT-TLS work for the specified connection. The specified connection will be reset and closed.

**Connection Close**

The specified secure connection is being closed.
**Connection Stop**
The specified secure connection is being stopped.

**Environment Create**
The specified AT-TLS environment is being created in the AT-TLS group to support the specified connection.

**Environment Delete**
The specified AT-TLS environment is no longer needed and is being deleted from the AT-TLS group. If this AT-TLS environment is not linked to a master AT-TLS environment, the corresponding System SSL environment will be closed.

**Initial Handshake**
The initial SSL handshake process is beginning for the specified connection.

**Reset Cipher Request**
A request to renegotiate the cipher is being processed.

**Reset Session Request**
A request to reset the session is being processed.

**SSL Control Data Read**
Control data, such as handshake or alert data, is being read.

*grp_act* is the name of the TTLSGroupAction.

*env_act* is the name of the TTLSEnvironmentAction. If a TTLSEnvironmentAction is not specified on the TTLSRule statement or if the event is not environment or connection related, this value will be *N/A*.

*conn_act* is the name of the TTLSConnectionAction. If a TTLSConnectionAction is not specified on the TTLSRule statement or if the event is not connection related, this value will be *N/A*.

*add_data* is additional information traced with the following events:

**Connection Abend Close**
The abend code.

**Initial Handshake**
The handshake role and, if handshake role is ServerWithClientAuth, the client authentication type.

**System action**
None.

**Operator response**
None.

**System programmer response**
If *event* is **Connection Abend Close**, then contact IBM software support services. If *event* is anything else, then there is no action to be taken.

**Module**
EZBTLMSG

**Procedure name**
EZBTLMSG

**EZD1283I**  TTLS Event GRPID: *gid* ENVID: *eid* CONNID: *cid* RC: *rcode* event *add_data*

**Explanation**
Application Transparent Transport Layer Security (AT-TLS) has finished the specified event. The message creation time and owning TCP/IP jobname of the process creating the message are included in the syslog trace.
prior to the message ID. This message has a syslog priority of DEBUG and is written to the syslog when AT-TLS trace option EVENT (8) is specified.

**gid** is a hexadecimal value that uniquely identifies the AT-TLS group supporting the connection or SSL environment.

**eid** is a hexadecimal value that uniquely identifies the AT-TLS environment supporting the connection. Multiple AT-TLS secure environments may be represented by a single master System SSL secure environment. If the **eid** is 00000000, the event does not apply to a specific environment.

**cid** is a hexadecimal value that uniquely identifies this TCP connection for the life of the connection. A previously issued message EZD1281I provides additional information about the connection. If the **cid** is 00000000, the event does not apply to a specific connection.

**rcode** is a System SSL or AT-TLS return code. A value other than zero indicates why the event failed. **rcode** values under 5000 are generated by System SSL and are defined in z/OS Cryptographic Services System SSL Programming. Rcode values over 5000 are generated by AT-TLS and are defined in Diagnosing AT-TLS problems in z/OS Communications Server: IP Diagnosis Guide.

**event** is an AT-TLS event. Possible values are:

- **Connection Abend Close**  
  An abend occurred while executing AT-TLS work for the specified connection. The specified connection will be reset and closed.

- **Connection Close**  
  The specified secure connection was closed.

- **Connection Init**  
  A secure connection was initiated for the specified connection.

- **Connection Stop**  
  The specified secure connection was stopped.

- **Data Decryption**  
  Application data was decrypted by System SSL.

- **Environment Close**  
  The specified AT-TLS environment was deleted. The corresponding System SSL environment was closed.

- **Environment Init**  
  A System SSL environment was initialized for the specified AT-TLS environment.

- **Environment Link**  
  The specified newly created AT-TLS environment could use an existing System SSL environment. The new AT-TLS environment was linked to a master AT-TLS environment that represents a single System SSL environment.

- **Environment Link Delete**  
  The specified linked AT-TLS environment was no longer needed. It was linked to a master AT-TLS environment. The AT-TLS environment and its link were deleted. If this was the last AT-TLS environment linked to the master AT-TLS environment, the master was also deleted and the corresponding System SSL environment will be closed.

- **Environment Master Close**  
  The specified master AT-TLS environment no longer had any linked AT-TLS environments. The corresponding System SSL environment was closed.

- **Environment Master Create**  
  The specified newly created AT-TLS environment could be linked to a master AT-TLS environment. The master AT-TLS environment did not exist yet and was created. The master AT-TLS environment corresponds to a single System SSL environment.

- **Environment Master Delete**  
  The specified master AT-TLS environment was no longer needed and was deleted from the AT-TLS group. The corresponding System SSL environment will be closed.

- **Environment Master Init**  
  A System SSL environment was initialized for the specified master AT-TLS environment.
Initial Handshake
The initial SSL handshake process was completed for the specified connection.

Messages Dropped
Syslog messages were dropped as a result of an excessive backlog of messages to be logged.

Policy Mapping
Policy map was attempted for the specified connection.

Received FIN
A TCP header with the FIN flag turned on was received on the secure connection.

Received Reset
A TCP header with the Reset flag turned on was received on the secure connection. The connection is closed.

Reset Cipher Request
A request to renegotiate the cipher was processed.

Reset Ignored
A TCP header with the Reset flag turned on was received in a response to an SSL close alert. The reset has been ignored.

Reset Session Request
A request to reset the SSL session ID was processed.

SSL Control Data Read
Control data, such as handshake or alert data, was read.

add_data is additional information traced with the following events:

Connection Abend Close
• The handle of the System SSL connection block.
• The handle of the System SSL environment, if used by the connection.
• The abend code.

Connection Close
• The handle of the System SSL connection block.
• The handle of the System SSL environment, if used by the connection.

Data Decryption
The handle of the System SSL connection block.

Environment Close
The handle of the System SSL environment representing the AT-TLS environment.

Environment Init
The handle of the System SSL environment representing the AT-TLS environment.

Environment Link
• The handle of the System SSL environment that represents the master AT-TLS environment.
• The master AT-TLS environment ID.

Environment Link Delete
• The handle of the System SSL environment that represents the master AT-TLS environment.
• The master AT-TLS environment ID.

Environment Master Close
The handle of the System SSL environment representing the master AT-TLS environment.

Environment Master Create
The AT-TLS environment ID just created that necessitates creation of this master AT-TLS environment.

Environment Master Delete
The handle of the System SSL environment representing the master AT-TLS environment.
Environment Master Init
The handle of the System SSL environment representing the master AT-TLS environment.

Initial Handshake
• The handle of the System SSL connection block.
• The handle of the System SSL environment used by the connection.
• The security protocol used by the secure connection.
• The cipher used by the secure connection.

Messages Dropped
The number of messages that were dropped.

Received FIN
The number of bytes of encrypted data on the AT-TLS receive queue when the FIN was received.

Reset Cipher Request
• The handle of the System SSL connection block.
• The handle of the System SSL environment used by the connection.

Reset Session Request
• The handle of the System SSL connection block.
• The handle of the System SSL environment used by the connection.

SSL Control Data Read
• The handle of the System SSL connection block.
• The handle of the System SSL environment used by the connection.

System action
None.

Operator response
None.

System programmer response
System Programmer response varies by event type:

If event is Connection Abend Close, then contact IBM software support services. For all other event values, if rcode has a value other than zero, use it to determine the cause of the error and correct if necessary. See z/OS Communications Server: IP Diagnosis Guide for more information.

Module
EZBTLMSG

Procedure name
EZBTLMSG

EZD1284I TTLS Flow Grpid: gid Envid: eid Connid: cid Rcode: rcode Action func_or_parm- add_data

Explanation
Application Transparent Transport Layer Security (AT-TLS) called System SSL to perform the specified action on the specified function or parameter. The message creation time and owning TCP/IP job name of the process creating the message are included in the syslog trace prior to the message ID. This message has a syslog priority
of DEBUG and is written to the syslog when AT-TLS trace option FLOW (16) is specified. Flow messages might follow the Start message and precede Event messages, depending on the trace options chosen.

*gid* is a hexadecimal value that uniquely identifies the AT-TLS group supporting the connection or SSL environment.

*eid* is a hexadecimal value that uniquely identifies the AT-TLS environment supporting the connection. Multiple AT-TLS environments might be represented by a single master System SSL secure environment. If the *eid* is 00000000, the event does not apply to a specific environment.

*cid* is a hexadecimal value that uniquely identifies this TCP connection for the life of the connection. A previously issued message EZD1281I provides additional information about the connection. If the *cid* is 00000000, the event does not apply to a specific connection.

*rcode* is a System SSL or AT-TLS return code. A value other than zero indicates why the event failed. *rcode* values under 5000 are generated by System SSL and are defined in *z/OS Cryptographic Services System SSL Programming*. *rcode* values over 5000 are generated by AT-TLS and are defined in *Diagnosing AT-TLS problems in z/OS Communications Server: IP Diagnosis Guide*.

*action* is the action that was taken. Possible values are:

**Call**
- Call a System SSL function.

**Get**
- Get a System SSL parameter value.

**Set**
- Set a System SSL parameter value.

*func_or_parm* is the name and enumeration value of the System SSL function that was called or the System SSL parameter whose value was set or retrieved. See *z/OS Cryptographic Services System SSL Programming* for more information about System SSL functions and parameters.

*add_data* is additional information traced with the following actions:

**Call**
- The System SSL handle for the Environment or Connection processed by the called function.

**Get**
- The value received for the parameter on the Get.

**Set**
- The value specified for the parameter on the Set.

**System action**
None.

**Operator response**
None.

**System programmer response**
If *rcode* has a value other than zero, use it to determine the cause of the error and correct if necessary. See *z/OS Communications Server: IP Diagnosis Guide* for more information.

**Module**

EZBTLMSG

**Automation**
Not applicable.
Procedure name
EZBTLMSG

Example

```
EZD1284I TTLS Flow GRPID: 00000001 ENVID: 00000004 CONNID: 0000018C RC: 0
Set GSK_KEYRING_FILE(201) - ibmuser_ring
```

**EZD1285I**  TTLS Data CONNID: *cid* dir Cipher data

**Explanation**

Application Transparent Transport Layer Security (AT-TLS) has sent or received non-application data on the specified connection. The message creation time and owning TCP/IP jobname of the process creating the message are included in the syslog trace prior to the message ID. This message has a syslog priority of DEBUG and is written to syslog when AT-TLS trace option DATA (32) is specified.

*cid* is a hexadecimal value that uniquely identifies this TCP connection for the life of the connection. A previously issued message EZD1281I provides additional information about the connection. If the *cid* is 00000000, the event does not apply to a specific connection.

*dir* is either of the following:

- **Recv**  
  AT-TLS has received data from the network on the connection.

- **Send**  
  AT-TLS is sending data to the network on the connection.

*data* is the System SSL control data being transferred.

**System action**

None.

**Operator response**

None.

**System programmer response**

None.

**Module**

EZBTLMSG

**Procedure name**

EZBTLMSG

**EZD1286I**  TTLS Error GRPID: *gid* ENVID: *eid* CONNID: *cid* LOCAL: *loc_ip*..*loc_port* REMOTE: *rem_ip*..*rem_port* JOBNAME: *jobname* USERID: *userid* RULE: *rule* RC: *rcode* event

**Explanation**

Application Transparent Transport Layer Security (AT-TLS) detected an error during the specified AT-TLS event. The message creation time and owning TCP/IP job name of the process creating the message are included in the syslog trace prior to the message ID. This message has a syslog priority of ERROR and is written to the syslog when AT-TLS trace option ERROR (2) is specified.

In the message text:
**gid**
The hexadecimal value that uniquely identifies the AT-TLS group supporting the connection or SSL environment.

**eid**
The hexadecimal value which uniquely identifies the AT-TLS environment supporting the connection. Multiple AT-TLS environments may be represented by a single master System SSL secure environment. If the **eid** is 00000000, the event does not apply to a specific environment.

**cid**
The hexadecimal value which uniquely identifies this TCP connection for the life of the connection. A previously issued message EZD1281I provides additional information about the connection. If the **cid** is 00000000, the event does not apply to a specific connection.

**loc_ip**
The local IPv4 or IPv6 address.

**loc_port**
The local port number.

**rem_ip**
The remote IPv4 or IPv6 address.

**rem_port**
The remote port number.

**jobname**
The job name of the application associated with this connection.

**userid**
The user ID of the application associated with this connection.

**rule**
The name of the TTLSRule statement that mapped this connection.

**rcode**
The System SSL or AT-TLS return code that indicates why the event failed. **rcode** values under 5000 are generated by System SSL and are defined in z/OS Cryptographic Services System SSL Programming. **rcode** values over 5000 are generated by AT-TLS and are defined in AT-TLS return codes in z/OS Communications Server: IP Diagnosis Guide.

**event**
The AT-TLS event that was in process when the error occurred. Possible values are:

- **Connection Abend Close**
  An abend occurred while executing AT-TLS work for the specified connection.

- **Connection Close**
  The specified secure connection was being closed.

- **Connection Init**
  A secure connection was being initiated for the specified connection.

- **Connection Stop**
  The specified secure connection was being stopped.

- **Data Decryption**
  Application data was being decrypted by System SSL.

- **Data Encryption**
  Application data was being encrypted by System SSL.

- **Environment Close**
  The specified AT-TLS environment was being deleted and the corresponding System SSL environment was being closed.

- **Environment Init**
  A System SSL environment was being initialized for the specified AT-TLS environment.
Environment Link
The specified newly created AT-TLS environment could use an existing System SSL environment. The new AT-TLS environment was being linked to a master AT-TLS environment that represents a single System SSL environment.

Environment Link Delete
The specified linked AT-TLS environment was no longer needed. It was linked to a master AT-TLS environment. The AT-TLS environment and its link were being deleted. If this is the last AT-TLS environment linked to the master AT-TLS environment, the master was also being deleted and the corresponding System SSL environment will be closed.

Environment Master Close
The specified master AT-TLS environment no longer had any linked AT-TLS environments. The corresponding System SSL environment was being closed.

Environment Master Create
The specified newly created AT-TLS environment could be linked to a master AT-TLS environment. The master AT-TLS environment did not exist yet and was being created. A master AT-TLS environment corresponds to a single System SSL environment.

Environment Master Delete
The specified master AT-TLS environment was no longer needed and was being deleted from the AT-TLS group. The corresponding System SSL environment will be closed.

Environment Master Init
A System SSL environment was being initialized for the specified master AT-TLS environment.

HandshakeTimeout Expired
A secure connection was being initialized, but did not complete in the HandshakeTimeout interval.

Initial Handshake
The initial SSL handshake was in process for the connection.

Policy Mapping
Policy was being mapped for the specified connection.

Reset Cipher Request
A request to renegotiate the cipher was being processed.

Reset Session Request
A request to reset the session was being processed.

SSL Control Data Read
Control data, such as handshake or alert data, was being read.

System action
None.

Operator response
None.

System programmer response
Use the rcode value to determine the cause of the error and correct if necessary. See z/OS Communications Server: IP Diagnosis Guide for more information.

Module
EZBTLMSG
Example


Procedure name

EZBTLMSG


Explanation

Application Transparent Transport Layer Security (AT-TLS) detected an error during the specified AT-TLS event. This message is written to the joblog when AT-TLS trace option JOBLOG (1) is specified and has the same information as EZD1286I, which goes to the syslog when AT-TLS trace option ERROR (2) is specified. See EZD1286I for more information.

System action

None.

Operator response

None.

System programmer response

Use the rcode value to determine the cause of the error and correct if necessary. See z/OS Communications Server: IP Diagnosis Guide for more information.

Module

EZBTLMSG

Example


Procedure name

EZBTLMSG

EZD1288I  Tcpname AT-TLS GROUP group_name INITIALIZATION FAILED - reason

Explanation

The initialization of the AT-TLS GROUP group_name did not complete successfully because of the reason specified.

In the message text:

Tcpname

The name of the TCPIP stack

group_name

The name of the AT-TLS group specified on a TTLSGroupAction statement
**reason**
Additional information about why the initialization of the group failed. *reason* is one of the following:

ICSF UNAVAILABLE FOR FIPS140 MODE GROUP – ICSF was not active when *tcpname* initialized the AT-TLS group *group_name*, which is configured with FIPS140 On. ICSF is required for FIPS140 support.

**System action**
TCPIP continues. The AT-TLS group *group_name* is marked as failed. Any connections attempting to use that group will be reset. AT-TLS sets a return code of 5018 for those connections.

**Operator response**
Contact the system programmer.

**System programmer response**
Use the *reason* to resolve the initialization failure.

For *reason* ICSF UNAVAILABLE FOR FIPS140 MODE GROUP:

- If the AT-TLS group needs to be FIPS140 enabled, start ICSF before installing the AT-TLS policy. To attempt starting the AT-TLS group again after ICSF is active, refresh the AT-TLS policy as described in **Action refresh** in z/OS Communications Server: IP Configuration Guide.
- If the AT-TLS group does not need to be FIPS140 enabled, specify FIPS140 Off on the TTLSGroupAction statement. Refresh the AT-TLS policy as described in **Action refresh** in z/OS Communications Server: IP Configuration Guide.

**User response**
None.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBTLMST

**Routing code**
2, 8

**Descriptor code**
12

**Automation**
This message goes to the console. Automation can notify the system programmer.

**Example**
```
EZD1288I TCPCS AT-TLS GROUP grp_act1 INITIALIZATION FAILED - ICSF UNAVAILABLE FOR FIPS140 MODE GROUP
```

```
EZD1289I *Tcpname* ICSF SERVICES ARE CURRENTLY AVAILABLE FOR AT-TLS GROUP *group_name*
```
**Explanation**
AT-TLS is starting AT-TLS group `group_name` and ICSF is active.

In the message text:

*Tcpname*
The name of the TCPIP stack

*group_name*
The name of the AT-TLS group specified on a TTLSGroupAction statement

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
None.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBTLMST

**Routing code**
2, 8

**Descriptor code**
12

**Automation**
Not applicable for automation.

**Example**

```
EZD1289I  Tcpname ICSF SERVICES ARE CURRENTLY AVAILABLE FOR AT-TLS GROUP group_name
```

```
EZD1290I  Tcpname ICSF SERVICES ARE CURRENTLY UNAVAILABLE FOR AT-TLS GROUP group_name
```

**Explanation**

AT-TLS is starting AT-TLS group `group_name` and ICSF is not active. ICSF services are not available for the AT-TLS group. This includes access to the cryptographic hardware, certificate keys stored in ICSF, Elliptical Curve Cryptography, and AES-GCM ciphers.
In the message text:

**Tcpname**
The name of the TCPIP stack

**group_name**
The name of the AT-TLS group specified on a TTLSGroupAction statement

**System action**
Processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
If ICSF is not being used, no response is required. If ICSF services are required, the AT-TLS group must be restarted when ICSF is available. After ICSF is active, refresh the AT-TLS policy as described in Action refresh in z/OS Communications Server: IP Configuration Guide.

**User response**
None.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBTLMST

**Routing code**
2, 8

**Descriptor code**
12

**Automation**
This message goes to the console. Automation can notify the system programmer if ICSF services are being used by AT-TLS groups.

**Example**

```
EZD1290I TCPCS ICSF SERVICES ARE CURRENTLY UNAVAILABLE FOR AT-TLS GROUP grp_act1

EZD1291I Active EXPLICITBINDPORTRANGE changed to begin_port - end_port by tcp_name on mvs_name
```
Explanation
The explicit bind port range, used to coordinate sysplex port allocation across the sysplex for applications issuing bind INADDR_ANY and IN6ADDR_ANY with port 0, has been changed by the specified stack on the specified MVS system.

This message is displayed only on the MVS system for the stack that caused the range to change.

In the message text:

begin_port
The first port in the new range.

end_port
The last port in the new range.

tcp_name
The name of the TCP/IP stack that caused the port range to change.

mvs_name
The name of the MVS system on which the TCP/IP stack is running.

System action
TCP/IP processing continues. Explicit bind sysplex ports will be allocated from the new range.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFEBR, EZBXFSCF

Routing code
11

Descriptor code
6

Example

EZD1291I  Active EXPLICITBINDPORTRANGE changed to 06000 - 06999 by TCP1 on MVS1

EZD1292I  No active EXPLICITBINDPORTRANGE is available from this stack
**Explanation**

This message is issued in response to a DISPLAY TCPIP,tcpname,SYSPLEX,PORTS command. The active explicit bind port range is not available to be displayed from this stack either because explicit bind port range processing is not enabled for this stack or this stack does not have access to the sysplexports coupling facility structure.

**System action**

TCP/IP continues

**Operator response**

Issue the DISPLAY TCPIP,tcpname,SYSPLEX,PORTS command, specifying a stack that has explicit bind port range processing enabled, or issue a DISPLAY NET,STATS,TYPE=CFS,STRNAME=sysplexports_structure_name command on a VTAM node that is connected to the sysplexports structure. See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for information about enabling the EXPLICITBINDPORTRANGE parameter. See the information about the DISPLAY STATS command information in z/OS Communications Server: SNA Operation for information about displaying sysplexport structure data.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZBXFIO2

**Routing code**

Not applicable.

**Descriptor code**

Not applicable.

**Example**

Not applicable.

**EZD1293I** Configured EXPLICITBINDPORTRANGE: begin_port - end_port

**Explanation**

This message is issued in response to a DISPLAY TCPIP,tcpname,SYSPLEX,PORTS command. It displays the explicit bind port range configured by specifying EXPLICITBINDPORTRANGE on the GLOBALCONFIG statement. The explicit bind port range configured on this stack might not be the range that is actively in use for allocating explicit bind sysplexports. Message EZD1294I displays the currently active explicit bind port range. See the
information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for information about specifying the EXPLICITBINDPORTRANGE parameter.

In the message text:

`begin_port`

The first port in the configured explicit bind port range.

`end_port`

The last port in the configured explicit bind port range.

**System action**

TCP/IP processing continues.

**Operator response**

None.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZBXFIO2

**Routing code**

Not applicable.

**Descriptor code**

Not applicable.

**Example**

```plaintext
EZD1293I  Configured EXPLICITBINDPORTRANGE: 06000 - 06999
```

**EZD1294I**  Active EXPLICITBINDPORTRANGE: `begin_port` - `end_port`

**Explanation**

This message is issued in response to a DISPLAY TCPIP,tcpname,SYSLEX,PORTS command. It displays the explicit bind port range that is actively in use in the sysplex (or subplex) that allocates explicit bind sysplexports.

In the message text:

`begin_port`

The first port in the active explicit bind port range.
**end_port**
The last port in the active explicit bind port range.

**System action**
TCP/IP continues.

**Operator response**
If the active explicit bind port range does not match the configured range for this stack (displayed in message EZD1293I), contact the system programmer.

**System programmer response**
Configure the same explicit bind port range on all the TCP/IP stacks that participate in explicit bind port range processing in the sysplex (or subplex). See the information about the `GLOBALCONFIG` statement in z/OS Communications Server: IP Configuration Reference for information about how to coordinate explicit bind port range configuration across stacks in the sysplex.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBXFIO2

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Example**

```
EZD1294I Active EXPLICITBINDPORTRANGE: 06000 - 06999
```

**EZD1295I**
No EXPLICITBINDPORTRANGE is configured on this stack

**Explanation**
This message is issued in response to a DISPLAY TCPIP,tcpname,SYSPLEX,PORTS command. The explicit bind port range is not configured on this stack, either because no `GLOBALCONFIG` statement was processed that specified the `EXPLICITBINDPORTRANGE` parameter, or a `GLOBALCONFIG` statement with the `NOEXPLICITBINDPORTRANGE` parameter was processed.

**System action**
TCP/IP continues.
Operator response
If you want this stack to participate in explicit bind port range processing, contact the system programmer.

System programmer response
See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for information about enabling the EXPLICITBINDPORTRANGE option.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFIO2

Routing code
Not applicable.

Descriptor code
Not applicable.

Example
Not applicable.

E镀锌1296I  EXPLICITBINDPORTRANGE exhausted

Explanation
There are no more ports available in the active range to be allocated for explicit bind port range processing. This message indicates that the explicit bind port range is too small to accommodate all the applications that issue bind requests to INADDR_ANY and IN6ADDR_ANY and port 0. This message is issued only once every 5 minutes when the TCP/IP stack attempts to allocate a new port from the explicit bind port range and finds that no ports are available. Display the active explicit bind port range by issuing a DISPLAY TCPIP,,SYSPLEX,PORTS command. See the information about DISPLAY TCPIP,,SYSPLEX in z/OS Communications Server: IP System Administrator's Commands.

System action
TCP/IP continues. Applications that issue bind requests to INADDR_ANY and IN6ADDR_ANY , port 0 on this stack will have their subsequent connections fail if a distributed DVIPA is chosen as a source IP address.

Operator response
Contact the system programmer.
**System programmer response**

Either extend the active explicit bind port range or configure a new explicit bind port range with enough ports to accommodate the way the range is used in your sysplex (or subplex). The increase in size for the range should be at least 64 times the number of TCP/IP stacks configured with the GLOBALCONFIG EXPLICITBINDPORTRANGE option in the sysplex (or subplex). For example, if your current range is 06000–06639 and you have five TCP/IP stacks configured with the GLOBALCONFIG EXPLICITBINDPORTRANGE statement, you probably want to increase the range by 320 ports to 06000–06959. See the information about the GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference for information about setting the EXPLICITBINDPORTRANGE statement. See the information about diagnosing network security services (NSS) server problems in z/OS Communications Server: IP Diagnosis Guide for information about how to determine a range for explicit bind ports.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZBTCNET

**Routing code**

2,8

**Descriptor code**

12

**Automation**

This is a system console message. You might automate on this message to determine when the EXPLICITBINDPORTRANGE is exhausted, so that you can extend the range, as described in the System Programmer Response.

**Example**

Not applicable.

**EZD1297I tcp_name is unable to set EXPLICITBINDPORTRANGE in the sysplexports structure**

**Explanation**

An attempt to set the explicit bind port range in the sysplexports coupling facility structure failed. This attempt was made during the processing of a GLOBALCONFIG statement with the EXPLICITBINDPORTRANGE parameter specified. The failure might be caused by a loss of access to the sysplexports structure, either as a result of a structure rebuild or a structure disconnect.

In the message text:

**tcp_name**

The name of the TCP/IP stack.
System action
TCP/IP processing continues. No explicit bind port range processing is performed on this stack.

Operator response
In the console log, check for any failure or rebuild messages referencing the sysplexports structure (the sysplexports structure name will be in the format EZBEPORTvvtt, where the vv value is the VTAM XCF group ID specified as a VTAM start option and the tt value is the TCP XCF group ID specified in the TCP profile on the GLOBALCONFIG statement.) If a structure rebuild was in process for the sysplexports structure used by this stack, wait for the rebuild to complete and issue a VARY TCPIP,,OBEYFILE command specifying a file containing the GLOBALCONFIG EXPLICITBINDPORTRANGE statement. If VTAM lost connectivity to the structure, issue the VARY NET,CFS,ACTION=CONNECT,STRNAME=structure_name command to re-establish connectivity to the structure. When connectivity is re-established, issue a VARY TCPIP,,OBEYFILE command specifying a file containing the GLOBALCONFIG EXPLICITBINDPORTRANGE statement. If neither of the previous conditions are true, contact the system programmer.

System programmer response
Take a dump of TCP stack address space, the VTAM address space, and the sysplexports structure and contact the IBM Software Support Center.

User response
Not applicable

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFUT6

Routing code
2,8

Descriptor code
12

Example
EZD1297I TCP1 is unable to set EXPLICITBINDPORTRANGE in the sysplexports structure

EZD1298I DYNAMIC VIPA dvipa DELETED FROM tcpstackname

Explanation
A DVIPA was deleted from the TCP/IP stack. The stack is no longer servicing any connections for this DVIPA.

In the message text:

dvipa
   The dynamic VIPA that was deleted.
**tcpstackname**
The name of the TCP/IP stack.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: DVIPA

**Module**
EZBXFDVI, EZBX6DVI

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

```
EZD1298I DYNAMIC VIPA 19.10.1.10 DELETED FROM TCPCS
EZD1299I VIPADISTRIBUTE keyword IS NOT VALID BECAUSE THE CURRENT DISTRIBUTION METHOD IS NOT SERVERWLM
```

**Explanation**
The DISTMETHOD keyword was not specified on the VIPADISTRIBUTE statement that is being processed. Either the distribution method was set to the default value BASEWLM, or a previous VIPADISTRIBUTE statement for the same DVIPA and port defined a distribution method other than SERVERWLM. The specified keyword is valid only when the distribution method is SERVERWLM.

This message appears only in the system log and is preceded by either message EZZ8469I or message EZZ8470I, which identifies the VIPADISTRIBUTE definition.

In the message text:
keyword
The keyword on the VIPADISTRIBUTE profile statement.

System action
TCP/IP continues. The VIPADISTRIBUTE statement with the keyword that is not valid is ignored.

Operator response
Contact the system programmer.

System programmer response
Remove the incorrect VIPADISTRIBUTE statement from the VIPADYNAMIC block, or correct the VIPADISTRIBUTE statement by deleting the parameter and keyword that is not valid, or by adding the DISTMETHOD parameter with the value SERVERWLM to change the distribution method. Then issue a VARY TCPIP,OBEYFILE command with an obey file that contains the corrected VIPADYNAMIC block. See the information about the VIPADYNAMIC statement in z/OS Communications Server: IP Configuration Reference for more information.

User response
Not applicable.

Problem determination
Not Applicable

Source
z/OS Communications Server TCP/IP: Configuration & Initialization

Module
ezbxfdv2, ezbx6dv2

Routing code
11

Descriptor code
6

Automation
Not applicable.

Example

```
EZD1299I VIPADISTRIBUTE PROCXCOST IS NOT VALID BECAUSE THE CURRENT DISTRIBUTION METHOD IS NOT SERVERWLM
```

```
EZD1300I TIER PARAMETER DOES NOT MATCH VIPADEFINE
```

Explanation
This message contains additional information for message EZZ8469I or EZZ8471I. The VIPADISTRIBUTE statement and the corresponding VIPADEFINE statement for this DVIPA must have the same TIER parameter value (either TIER1, TIER2, or not specified), but the TIER parameter values do not match.
System action
TCP/IP continues. The VIPADISTRIBUTE statement is rejected.

Operator response
Contact the system programmer.

System programmer response
Issue the Netstat VIPADCFG/-F command to determine which DVIPAs are configured on this stack.

If the VIPADISTRIBUTE statement was incorrect, change that statement in a VARY TCPIP,,OBEYFILE data set to specify the appropriate TIER parameter value. If the VIPADEFINExE statement was incorrect, specify a VIPADELETe statement for the DVIPA in a data set that is referenced by a VARY TCPIP,,OBEYFILE command, followed by a VIPADEFINExE statement for that DVIPA that specifies the appropriate TIER parameter. Also, to delete any existing VIPADISTRIBUTE DEFINE statements that reference this DVIPA, include the VIPADISTRIBUTE DELETE statements before the VIPADELETe statement in the data set.

See the information about the VIPADYNA믹 statement in z/OS Communications Server: IP Configuration Reference for more information about the VIPADISTRIBUTE profile statement.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFDV2

Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example
Not applicable.

EZD1301I VIPA IS DEFINED WITH THE CPCSCOPE PARAMETER BUT WITHOUT THE TIER2 PARAMETER

Explanation
This message contains additional information for message EZZ8469I. A VIPADISTRIBUTE DEFINE statement appears in a TCP/IP profile or VARY TCPIP,,OBEYFILE data set. The VIPA that it references is defined with the CPCSCOPE parameter, but without the TIER2 parameter. These DVIPAs cannot be distributed.
System action
TCP/IP continues. The VIPADISTRIBUTE statement is rejected.

Operator response
Contact the system programmer.

System programmer response
Issue the Netstat VIPADCFG/-F command to determine which DVIPAs are configured on this stack. If the VIPADISTRIBUTE DEFINE statement was incorrect, change that statement in a VARY TCPIP,OBEYFILE data set to use a DVIPA that does not specify the CPCSCOPE parameter or that specifies the CPCSCOPE and TIER2 parameters. If the VIPADEFINE statement was incorrect, specify a VIPADELETE statement for the DVIPA, followed by a VIPADEFINE statement for that DVIPA that does not specify the CPCSCOPE parameter or that specifies the CPCSCOPE and TIER2 parameters.

See the information about the VIPADYNAMIC statement in z/OS Communications Server: IP Configuration Reference for more information about the VIPADISTRIBUTE profile statement.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFDV2

Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example
Not applicable.

**EZD1302I** DVIPA ipaddr WITH CPCSCOPE IS ALREADY DEFINED ON A DIFFERENT CPC

Explanation
A DVIPA was defined by a VIPADEFINE or a VIPABACKUP statement in a TCP/IP profile or in a VARY TCPIP,OBEYFILE data set with the CPCSCOPE parameter specified, but that DVIPA has already been defined on a different central processor complex (CPC). Definitions of CPCSCOPE DVIPAs must remain within a CPC.

In the message text:
ipaddr
The IP address specified on the VIPDEFINE or VIPABACKUP statement.

System action
TCP/IP continues. The VIPDEFINE or VIPABACKUP statement is rejected.

Operator response
Contact the system programmer.

System programmer response
Move the VIPDEFINE or the VIPABACKUP statement to a TCP/IP profile that will be used by a stack that is on the same CPC as the original definition of this DVIPA or issue the VARY TCPIP,OBEYFILE command for a stack that is on the same CPC as the original definition of this DVIPA.

See the information about the VIPADYNAMIC statement in z/OS Communications Server: IP Configuration Reference for more information about the VIPDEFINE and VIPABACKUP profile statements.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFDV2

Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example
EZD1302I DVIPA 10.61.11.1 WITH CPCSCOPE IS ALREADY DEFINED ON A DIFFERENT CPC

EZD1303I GRE PARAMETER CANNOT BE MODIFIED FOR DVIPA ipaddr

Explanation
You tried to do one of the following:

- Specify a VIPADISTRIBUTE DEFINE statement with a GRE parameter but a VIPADISTRIBUTE statement was already specified without the GRE parameter.
• Specify a VIPADISTRIBUTE DEFINE statement without a GRE parameter but a VIPADISTRIBUTE statement was already specified with the GRE parameter.

If you want to specify the GRE parameter, it must be specified on the first VIPADISTRIBUTE statement that is specified for a DVIPA. If you specify GRE parameter on the first VIPADISTRIBUTE statement, all subsequent VIPADISTRIBUTE statements for that DVIPA must specify GRE.

In the message text:

ipaddr
   The IP address that is specified on the VIPADISTRIBUTE statement.

**System action**
TCP/IP continues. The VIPADISTRIBUTE statement is rejected.

**Operator response**
Contact the system programmer.

**System programmer response**
If GRE was specified on a VIPADISTRIBUTE DEFINE statement but a GRE parameter was not specified on an earlier definition, do one of the following:

• Correct the VIPADISTRIBUTE DEFINE statement by removing the GRE parameter.
• Enable GRE processing for distribution of this DVIPA by deleting all previous VIPADISTRIBUTE statements for this DVIPA that did not specify GRE and then reissuing the VIPADISTRIBUTE statements with the GRE parameter specified.

If GRE was not specified on a VIPADISTRIBUTE DEFINE statement but a GRE parameter was specified on an earlier definition, perform one of the following:

• Correct the VIPADISTRIBUTE DEFINE statement by adding the GRE parameter.
• Disable GRE processing for distribution of this DVIPA by deleting all previous VIPADISTRIBUTE statements for this DVIPA that specified GRE and then reissuing the VIPADISTRIBUTE statements without the GRE parameter specified.

See the information about the VIPADYNAMIC statement in z/OS Communications Server: IP Configuration Reference for more information about the VIPADISTRIBUTE profile statement.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBXFDV2

**Routing code**
2,8
**Example**

EZD1303I  GRE PARAMETER CANNOT BE MODIFIED FOR DVIPA 10.22.9.11

**EZD1305I  TARGCONTROLLED DISTMETHOD CANNOT BE ENABLED FOR DVIPA ipaddr**

**Explanation**

DISTMETHOD TARGCONTROLLED was found on a VIPADISTRIBUTE DEFINE statement for a DVIPA that already had a VIPADISTRIBUTE statement that specified a different distribution method. DISTMETHOD TARGCONTROLLED must be specified on the first VIPADISTRIBUTE statement that is specified for a DVIPA.

In the message text:

*ipaddr*

The IP address specified on the VIPADISTRIBUTE statement that contains the rejected DISTMETHOD parameter.

**System action**

TCP/IP continues. The VIPADISTRIBUTE statement is rejected.

**Operator response**

Contact the system programmer.

**System programmer response**

Do one of the following:

- Correct the VIPADISTRIBUTE DEFINE statement by removing the DISTMETHOD TARGCONTROLLED parameter.
- Enable TARGCONTROLLED as a distribution method, delete all previous VIPADISTRIBUTE statements for this DVIPA that specify a DISTMETHOD value other than TARGCONTROLLED, then reissue the VIPADISTRIBUTE statements with DISTMETHOD TARGCONTROLLED specified.

See the information about the VIPADYNAMIC statement in z/OS Communications Server: IP Configuration Reference for more information about the VIPADISTRIBUTE profile statement.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZBXFDV2
Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example

EZD1305I  TARGCONTROLLED DISTMETHOD CANNOT BE ENABLED FOR DVIPA 10.22.9.11

EZD1306I  TARGCONTROLLED DISTMETHOD CANNOT BE DISABLED FOR DVIPA ipaddr

Explanation
A DISTMETHOD value other than TARGCONTROLLED was found on a VIPADISTRIBUTE DEFINE statement for a DVIPA that already had a VIPADISTRIBUTE statement that specified DISTMETHOD TARGCONTROLLED. DISTMETHOD cannot be modified from TARGCONTROLLED to another distribution method on subsequent VIPADISTRIBUTE definitions for a DVIPA.

In the message text:

ipaddr  The IP address specified on the VIPADISTRIBUTE statement that contains the rejected DISTMETHOD parameter.

System action
TCP/IP continues. The VIPADISTRIBUTE statement is rejected.

Operator response
Contact the system programmer.

System programmer response
Do one of the following:

- Correct the VIPADISTRIBUTE DEFINE statement by changing the DISTMETHOD value to TARGCONTROLLED.
- Disable TARGCONTROLLED as a distribution method, delete all previous VIPADISTRIBUTE statements for this DVIPA that specify DISTMETHOD TARGCONTROLLED, then reissue the VIPADISTRIBUTE statements with a DISTMETHOD other than TARGCONTROLLED specified.

See the information about the VIPADYNAMIC statement in z/OS Communications Server: IP Configuration Reference for more information about the VIPADISTRIBUTE profile statement.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP
**Example**

EZD1306I  TARGCONTROLLED DISTMETHOD  CANNOT BE DISABLED FOR DVIPA 10.22.9.11

**EZD1307I**  CONTROLPORT VALUE CANNOT BE MODIFIED FOR DVIPA *ipaddr*

**Explanation**

There are two possible causes of this error:

- The CONTROLPORT parameter was found on a VIPADISTRIBUTE DEFINE statement for a DVIPA. The value of the parameter was different from the CONTROLPORT value that was currently in effect as a result of a previous VIPADISTRIBUTE DEFINE statement.
- The default CONTROLPORT value was used for the VIPADISTRIBUTE DEFINE statement for this DVIPA. The default value was different from the CONTROLPORT value that was currently in effect as a result of a previous VIPADISTRIBUTE DEFINE statement.

The CONTROLPORT value cannot be modified on subsequent VIPADISTRIBUTE definitions.

In the message text:

*ipaddr*

The IP address specified on the VIPADISTRIBUTE statement that contains the rejected CONTROLPORT parameter.

**System action**

TCP/IP continues. The VIPADISTRIBUTE statement is rejected.

**Operator response**

Contact the system programmer.

**System programmer response**

Do one of the following:

- Correct the VIPADISTRIBUTE DEFINE statement by modifying it to use the same CONTROLPORT value as used on previous VIPADISTRIBUTE DEFINE statements.
- Modify the CONTROLPORT value for the DVIPA, delete all previous VIPADISTRIBUTE statements that use a different CONTROLPORT value, then reissue the VIPADISTRIBUTE statements with the new CONTROLPORT value specified.

See the information about the VIPADYNAMIC statement in z/OS Communications Server: IP Configuration Reference for more information about the VIPADISTRIBUTE profile statement.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFDV2

Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example
EZD1307I CONTROLPORT VALUE CANNOT BE MODIFIED FOR DVIPA 10.22.9.11

EZD1308I DVIPA ipaddr IS ALREADY DEFINED WITH DIFFERENT TIER1, TIER2 OR CPCSCOPE PARAMETER

Explanation
A VIPDEFINE or VIPABACKUP statement appears in a TCP/IP profile or VARY TCPIP,,OBEYFILE file with the TIER1, TIER2, or CPCSCOPE parameter specified, but that DVIPA was already defined with a different TIER1, TIER2, or CPCSCOPE parameter specified. This message might also be issued if the VIPDEFINE or VIPABACKUP statement did not specify a TIER1, TIER2, or CPCSCOPE parameter, but the DVIPA was already defined with a TIER1, TIER2, or CPCSCOPE parameter specified.

In the message text:

ipaddr
The IP address specified on the VIPDEFINE or VIPABACKUP statement.

System action
TCP/IP continues. The VIPDEFINE or VIPABACKUP statement is rejected.

Operator response
Contact the system programmer.

System programmer response
If the initial VIPDEFINE or VIPABACKUP statement was incorrect, issue a VIPDELETE statement for the DVIPA, followed by a VIPDEFINE or VIPABACKUP statement for that DVIPA that specifies the appropriate TIER or CPCSCOPE parameter. If the current VIPDEFINE or VIPABACKUP statement is incorrect, change that
statement to specify the TIER or CPCSCOPE parameter that is specified on the previous definition, and restart the stack (if the TCP/IP profile was changed) or reissue the VARY TCPIP,OBEYFILE command.

See the information about the VIPADYNAMIC statement in z/OS Communications Server: IP Configuration Reference for more information about the VIPADEFIND and VIPABACKUP profile statement.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Configuration & Initialization

Module
EZBXFDVI

Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example

EZD1308I DVIPA 10.60.1.1 IS ALREADY DEFINED WITH DIFFERENT TIER1, TIER2 OR CPCSCOPE PARAMETER

EZD1309I DESTIP ALL CANNOT BE SPECIFIED WHEN GRE IS SPECIFIED

Explanation
This message contains additional information for message EZZ8471I. A DESTIP ALL parameter was found on a VIPADISTRIBUTE DELETE statement for a tier 1 DVIPA whose VIPADISTRIBUTE DEFINE statement specified a GRE parameter, but the DESTIP ALL parameter is not allowed for tier 1 DVIPAs when the GRE parameter is specified on the VIPADISTRIBUTE DEFINE statement. You must explicitly list the target IP addresses.

System action
TCP/IP continues. The VIPADISTRIBUTE statement is rejected.

Operator response
Contact the system programmer.

System programmer response
Change the VIPADISTRIBUTE DELETE statement to explicitly list the target IP addresses to be removed from distribution.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFDV2

Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example
Not applicable.

EZD1310I tcp_name DISCARDED INBOUND SYSPLEX DISTRIBUTED PACKETS

Explanation
An unusually high rate of inbound sysplex-distributed traffic resulted in severe contention for internal TCP/IP resources, as measured by the maximum concurrent service request block (SRB) threshold. TCP/IP discarded inbound sysplex-distributed traffic that is associated with the SRBs that exceeded the threshold to protect the system from exhausting critical resources. This message is issued once per sysplex monitor interval if the concurrent SRB threshold is exceeded. The message is deleted after an entire monitor interval passes without any SRBs exceeding the threshold.

In the message text:

 tcp_name
The name of the TCP/IP stack.

System action
TCP/IP continues.

Operator response
Contact the system programmer.

System programmer response
Investigate the cause of the surge in distributed traffic. Contact IBM software support services if the message persists.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Sysplex Distributor

Module
EZBXFPDM

Routing code
2,8

Descriptor code
3

Example
EZD1310I TCPCS DISCARDED INBOUND SYSPLEX DISTRIBUTED PACKETS

EZD1311I  SYSPLEX DISTRIBUTOR CONTROL CONNECTION TO DESTINATION IP ADDRESS ip_addr CLOSED DUE TO A PROTOCOL ERROR WITH RETURN CODE internal_retcode.

Explanation
The control connection to the non-z/OS target has been closed as the result of a protocol error. The non-z/OS target cannot be used as a target for distributed connections.

In the message text:
- **ip_addr**
  The IP address of a distributed non-z/OS target.
- **internal_retcode**
  The internal return code that was set by the function that detected the protocol error.

System action
TCP/IP continues.

Operator response
Contact the system programmer.

System programmer response
Contact the IBM software support services with the TCP/IP profile and the system log.

User response
Not applicable.

Problem determination
Not applicable.
Source
z/OS Communications Server TCP/IP: Sysplex Distributor

Module
EZBXFSUB

Routing code
2,8

Descriptor code
12

Automation
This message is written to the system console. This message is a good candidate for automation. Automation can allow you to monitor the status of sysplex distributor control connections to non-z/OS targets.

Example

<table>
<thead>
<tr>
<th>EZD1311I</th>
<th>SYSPLEX DISTRIBUTOR CONTROL CONNECTION TO DESTINATION IP ADDRESS 9.42.105.53 CLOSED DUE TO PROTOCOL ERROR WITH RETURN CODE 00000016</th>
</tr>
</thead>
</table>

| EZD1312I  | HEALTH CHECKER SETUP FAILED FOR resource_name MACRO macro_name RC rtn_code RSN rsn_code |

Explanation
The initialization of support for checks in the IBM Health Checker for z/OS failed for the specified system resource.

In the message text:

resource_name
The name of the system resource associated with the IBM Health Checker for z/OS initialization failure.
- If the failure involves a TCP/IP stack, resource_name is the job name of the TCP/IP stack.
- If the failure involves the system resolver, resource_name is RESOLVER.

macro_name
The name of the macro that failed. This value is CSVDYNEX or HZSCHECK.

rtn_code
The return code from the failing macro. It is displayed as 2 hexadecimal digits.

rsn_code
The reason code from the failing macro. It is displayed as 4 hexadecimal digits.

System action
The system resource continues initialization. IBM Health Checker for z/OS checks are not performed for the specified system resource.

Operator response
Contact the system programmer.
System programmer response

If the failing macro was CSVDYNEX, look up the \textit{rtn\_code} and \textit{rsn\_code} values in the CSVDYNEX macro description in z/OS MVS Programming: Authorized Assembler Services Reference ALE-DYN to determine the cause of the failure. If the failing macro was HZSCHECK, look up the \textit{rtn\_code} and \textit{rsn\_code} values in the HZSCHECK macro description in IBM Health Checker for z/OS: User’s Guide to determine the cause of the failure. When the cause of the failure has been determined, correct the problem and stop and restart the system resource to set up the appropriate checks in IBM Health Checker for z/OS.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communication Server TCP/IP or z/OS Communications Server Resolver

Module
EZBTIINI or EZBREINI

Routing code
2

Descriptor code
12

Example

\begin{verbatim}
EZD1312I HEALTH CHECKER SETUP FAILED FOR TCPCS1 MACRO CSVDYNEX RC 0C RSN 0C02
EZD1313I REQUIRED SAF SERVAUTH PROFILE NOT FOUND resource_name
\end{verbatim}

Explanation
TCP/IP denied a request because SAF indicated no decision (return code 4) for the requested resource name in the SERVAUTH class.

In the message text:

\textit{resource\_name}

The fully qualified name of the resource being checked.

System action
TCP/IP denies the request and processing continues.

Operator response
Save the system log and contact the system programmer.

System programmer response
The most likely cause of this error is that no profile was defined for this resource name. If you think that a profile was defined for this resource, ensure that the following are true:
1. RACF (or other SAF compliant security server) was started.
2. The SETROPTS CLASSACT(SERVAUTH) command was issued.
3. The SETROPTS GENERIC(SERVAUTH) command was issued before you defined any profiles that contain the asterisk (*) or percent sign (%) wildcard characters. Profiles entered before this command are not recognized by RACF as generic, but as discrete.
4. The spelling of the intended profile name matches the spelling of the resource name in the message.
5. The SETROPTS RACLST(SERVAUTH) command was issued or the SETROPTS RACLST(SERVAUTH) REFRESH command was issued after any changes were made.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: TCP/IP stack

**Module**

Not applicable.

**Routing code**

2,8

**Descriptor code**

12

**Example**

```
EZD1313I REQUIRED SAF SERVAUTH PROFILE NOT FOUND  EZB.INITSTACK.MVS001.TCPIP
EZD1318I  A CONNECTION FROM A client_type CLIENT HAS BEEN REJECTED BECAUSE THE NSS SERVER CAN ONLY HANDLE max_client CONCURRENT client_type2 CLIENTS
```

**Explanation**

The network security services daemon (NSSD) has reached the limit of the number of clients that can be supported for a given type of client. No additional connections will be accepted until the number of active connections for the type of client attempting to connect is less than the supported limit.

In the message text:

- **client_type**
  The possible values are Network Security Server or Network Management Interface.

- **max_client**
  The maximum number of clients of the specified type that the NSSD can service at any time.

- **client_type2**
  The same value as the client_type value.

**System action**

The new connection is closed and the server continues with its existing set of clients.
Operator response
Contact the system programmer.

System programmer response
See the information about NSS server capacity considerations in z/OS Communications Server: IP Configuration Guide for more information about the number of NSS server and network management interface clients that a single NSSD instance can support.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssClient.cpp, NmiClient.cpp

Routing code
10

Descriptor code
12

Example
EZD1318I A CONNECTION FROM A NETWORK SECURITY SERVER CLIENT HAS BEEN REJECTED BECAUSE THE NSS SERVER CAN ONLY HANDLE 500 CONCURRENT NETWORK SECURITY SERVER CLIENTS

EZD1319I ERROR (errno | errnojr | description) WHILE OPENING MESSAGE CATALOG name - DEFAULT MESSAGES WILL BE USED

Explanation
The network security services daemon (NSSD) was unable to open the message catalog. Default messages will be used.

In the message text:

errno
The UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

errnojr
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

description
Describes the meaning of the errno value.

name
The message catalog file name that the NSSD was attempting to open.
System action
The default message catalog will be used by the NSSD.

Operator response
Contact the system programmer.

System programmer response
Correct the error indicated by the erno, errojr, and description values. A common cause of this error message is an incorrectly set NSSD_FILE environment variable. See the Steps for configuring the NSS server in z/OS Communications Server: IP Configuration Guide for information about the NSSD_FILE environment variable.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssLog.cpp

Routing code
10

Descriptor code
12

Example
EZD1319 ERROR (129 | 053B006C | EDC5129I NO SUCH FILE OR DIRECTORY.) WHILE OPENING MESSAGE CATALOG NSSMSG.CAT - DEFAULT MESSAGES WILL BE USED

EZD1320I NSSD CTRACE PARMLIB MEMBER pname WAS NOT FOUND

Explanation
The network security services (NSS) server was unable to find the specified parmlib member and is initialized with the MINIMUM tracing option.

In the message text:

pname
The name of the CTRACE parmlib member.

System action
NSSD CTRACE initializes with the minimum tracing option; the NSS server continues.

Operator response
If different CTRACE options are required, contact the system programmer.
System programmer response
If different CTRACE options are required, configure the CTRACE parmlib member. See the information about the Sockets API traces in z/OS Communications Server: IP Diagnosis Guide for more information about configuring the CTrace parmlib member.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssTrace.cpp

Routing code
10

Descriptor code
12

Example

EZD1320I NSSD CTRACE PARMLIB MEMBER CTINSS00 WAS NOT FOUND
EZD1321I NSSD CONFIGURATION FILE WAS NOT SPECIFIED - USING DEFAULTS FOR ALL NSSD CONFIGURATION PARAMETERS

Explanation
No configuration file was specified, so the network security services daemon (NSSD) attempted to use the default configuration file. The default NSSD configuration file does not exist so the server uses the default configuration values.

System action
NSSD reverts to default values and processing continues.

Operator response
None.

System programmer response
If you want values other than the system supplied defaults, create a configuration file. See the information about the network security services (NSS) server information in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file.

User response
No action needed.
Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssReadConfiguration.cpp

Routing code
10

Descriptor code
12

Example
Not applicable.

EZD1322I   INTERNAL ERROR IN MODULE modid : errid | value1 | value2 | value3

Explanation
The network security services daemon (NSSD) detected an internal error. Additional diagnostic messages might be issued.

In the message text:

modid
An internal identifier that indicates the module that detected the error.

errid
An internal identifier for this error in the detecting module.

value1
Internal error information.

value2
Internal error information.

value3
Internal error information.

System action
Results are unpredictable. One or more address space dumps can be produced with dump titles that match the message text.

Operator response
Contact the system programmer.

System programmer response
Contact IBM software support services and provide a syslog that includes this message. If available, provide CTRACE information for component SYSTCPNS. If available, provide any dumps associated with this message.

User response
Not applicable.
Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
Numerous.

Routing code
10

Descriptor code
12

Example
EZD1322I INTERNAL ERROR IN MODULE MGRWLSV : 1 | 1 | 121 | 0
EZD1323I CERTIFICATE (label) DOES NOT CONTAIN AN ISSUER NAME

Explanation
A certificate in the network security services (NSS) server certificate repository does not contain an issuer field. The NSS server requires that certificates contain an issuer field as defined in RFC 2459. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

In the message text:

label
The certificate repository label that identifies the certificate that did not contain an issuer field.

System action
The certificate is ignored by the network security server daemon.

Operator response
None.

System programmer response
Remove the certificate from the NSS server certificate repository. If this was a certificate to be used on behalf of an NSS client, obtain a new certificate containing an Issuer field.

User response
None.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server
Module
CertRepository.cpp

Routing code
10

Descriptor code
12

Example

EZD1323I CERTIFICATE ( CLIENT1CERT ) DOES NOT CONTAIN AN ISSUER NAME

EZD1324I CERTIFICATE ( label ) CANNOT BE USED TO CREATE A SIGNATURE AND IS NOT A CERTIFICATE AUTHORITY CERTIFICATE

Explanation
A certificate in the network security server certificate repository cannot be used to create a digital signature and is not a Certificate Authority certificate. A certificate must contain a private key to be used to create a signature.

If a KeyUsage extension is present in the certificate the digitalSignature bit must be set.

To be a Certificate Authority certificate the certificate must meet one of the following conditions:

• The certificate contains a basic constraints extension that indicates that the subject of this certificate is a Certificate Authority.

• The certificate does not contain a basic constraints extension, but the certificate is marked as trusted in the certificate repository and the issuer name in the certificate is equal to the subject name in the certificate (that is, it is self signed).

In the message text:

label
The certificate repository label identifying the certificate.

System action
The certificate is ignored by the network security services daemon. Processing continues.

Operator response
None.

System programmer response
If this certificate is intended to be used on behalf of a network security services (NSS) client to create a signature, then verify that the private key is stored in the certificate repository. If the private key is not stored in the repository remove the certificate from the repository and add it back to the repository with its private key. If the KeyUsage extension is present in the certificate and the digitalSignature bit is not set, the certificate cannot be used to create a signature. A new certificate must be obtained. If this certificate is intended to be used as a Certificate Authority certificate and it is self-signed, then verify that it is marked as trusted. If it is not self-signed then the certificate cannot be used as a Certificate Authority certificate. A new certificate must be obtained.

User response
None.
Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
CertRepository.cpp

Routing code
10

Descriptor code
12

Example

EZD1324I CERTIFICATE ( CERTWITHNOKEY ) CANNOT BE USED TO CREATE A SIGNATURE AND IS NOT A CERTIFICATE AUTHORITY CERTIFICATE

EZD1325I  ERROR PROCESSING CERTIFICATE ( label | error: error )

Explanation
An unexpected error was encountered while processing a certificate in the network security services (NSS) server certificate repository. Additional diagnostic messages might be issued.

In the message text:

label
  The certificate repository label that identifies the certificate that was being processed when the error was encountered.

error
  An internal identifier for this error in the detecting module.

System action
The certificate is ignored by the network security services daemon (NSSD).

Operator response
None.

System programmer response
If the certificate is not intended to be used to create a signature or to be used as a certificate authority certificate, then remove the certificate from the certificate repository.

If the certificate is intended to be used to create a signature or used as a certificate authority certificate, then perform the actions in the problem determination.

User response
None.
**Problem determination**
If this problem persists, enable the CERTINFO syslog level in the NSSD configuration file and cause the certificate repository to be reprocessed. Check for additional messages relating to this certificate. The certificate repository can be reprocessed by stopping and restarting the NSSD or by modifying the certificate repository.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
CertRepository.cpp

**Routing code**
10

**Descriptor code**
12.

**Example**

```
EZD1325I  ERROR PROCESSING CERTIFICATE  ( LABEL:  BADCERT  |  ERROR:  001F )

EZD1326I  REQUEST TYPE request WITH CORRELATOR ID corrid FROM CLIENT clientname FAILED -
           RETURN CODE returncode  REASON CODE reasoncode
```

**Explanation**
A request from a network security services (NSS) client failed.

In the message text:

- **request**  
  An identifier that describes the type of request.

- **corrid**  
  A 16-byte identifier used by a client to uniquely identify a request sent to the NSS server.

- **clientname**  
  The name of the NSS client that originated the request.

- **returncode**  
  Documents the type of failure. These return codes are listed and described in the networking management information in z/OS Communications Server: IP Programmer's Guide and Reference.

- **reasoncode**  
  A reason code that documents the reason for the failure. See the information about network manager return and reason codes in z/OS Communications Server: IP Programmer's Guide and Reference.

**System action**
The error information is returned to the NSS client. The NSSD continues.

**Operator response**
Contact the system programmer.

**System programmer response**
See the information about diagnosing network security services (NSS) server problems in z/OS Communications Server: IP Diagnosis Guide to determine the appropriate response.
Problem determination
The NSS client might provide additional diagnostic messages containing the same correlator as message EZD1326I. If the administrator of the NSS client can provide diagnostic information, the matching correlator can be used to locate the specific failure condition in the NSS server log file.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
Various

Routing code
10

Descriptor code
12

Example
EZD1326I Request type NSS_CreateSignatureReqToSrv with correlator ID 000000000000020D0000000000000000 from client VIRAIPSECREG_TCPCS8_RGHT failed - return code EINVAL reason code NSSRsnBadLIDType

EZD1327I NSSD CTRACE INITIALIZATION ERROR - FUNCTION function RETURN CODE rc REASON CODE rsn

Explanation
The Network Security Services Daemon (NSSD) failed to initialize the CTRACE subsystem.

In the message text:

**function**
The function that is being processed when the CTRACE error occurred.

**rc**
The error return code.

**rsn**
The error reason code.

System action
NSSD continues without CTRACE enabled.

Operator response
See the information about CTRACE in z/OS MVS Programming: Authorized Assembler Services Reference ALE-DYN for the return code and reason code explanations for the different CTrace functions. See the information about CTRACE—RESOLVER in z/OS Communications Server: IP Diagnosis Guide for more information about diagnosing CTRACE problems. Ensure that storage is available for the size of the trace buffers. Ensure that CTRACE definition parameters are set correctly. If these checks do not reveal the cause of the problem, contact the system programmer.
System programmer response

If the problem cannot be resolved, contact IBM software support services and provide a syslog that includes this message.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP: Network Security Server

Module

NsTrace.cpp

Routing code

10

Descriptor code

12

Example

```
EZD1327I NSSD CTRACE INITIALIZATION ERROR : FUNCTION CTSSM RETURN CODE 00000004 REASON CODE 00000004
EZD1328I CERTIFICATE REPOSITORY name SUCCESSFULLY PROCESSED FOR event
```

Explanation

The network security services daemon (NSSD) successfully updated its internal representation of the certificate repository.

In the message text:

- **name**: The name of the network security services (NSS) server certificate repository.
- **event**: The event that caused the certificate repository to be updated. Possible values are:
  - NSS initialization
  - NSS MODIFY command
  - NSS certificate repository updated

System action

The NSS server uses the new certificate repository.

Operator response

None.
**System programmer response**
None.

**User response**
None.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
CertMgr.cpp

**Routing code**
10

**Descriptor code**
12

**Example**

```
EZD1328I CERTIFICATE REPOSITORY NSSD/KEYRING SUCCESSFULLY PROCESSED FOR NSS INITIALIZATION

EZD1329I ERROR ENCOUNTERED PROCESSING CERTIFICATE REPOSITORY name FOR event
```

**Explanation**
The network security services daemon (NSSD) was unsuccessful in updating its internal representation of the certificate repository.

In the message text:

- **name**
  The name of the network security services (NSS) server certificate repository.

- **event**
  The event that triggered the certificate repository to be updated. Possible values are:

  • NSS initialization
  • NSS MODIFY command
  • NSS certificate repository update

**System action**
The NSS server continues to use its previous internal representation of the certificate repository. If a previous internal representation of the certificate repository did not exist, the NSS server will not be able to process certificate requests from network security clients until action is taken to correct the problem.

**Operator response**
None.
System programmer response
Correct the problem and issue the NSSD MODIFY command.

User response
Not applicable.

Problem determination
If this problem persists, enable the CERTINFO syslog level in the NSSD configuration file and cause the certificate repository to be reprocessed. Check for additional messages relating to this certificate. The certificate repository can be reprocessed by stopping and restarting the NSSD or by modifying the certificate repository.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
CertMgr.cpp

Routing code
10

Descriptor code
12

Example

<table>
<thead>
<tr>
<th>EZD1329I</th>
<th>ERROR ENCOUNTERED PROCESSING CERTIFICATE REPOSITORY NSSD/KEYRING FOR NSS INITIALIZATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1330I</td>
<td>ERROR (gsk_rc</td>
</tr>
</tbody>
</table>

Explanation
The network security services daemon (NSSD) was unable to open the certificate repository specified on the NssConfig statement.

In the message text:

**gsk_rc**
The hexadecimal CMS status code. See the information about the CMS status codes in z/OS Cryptographic Services System SSL Programming.

**description**
Describes the meaning of the gsk_rc value.

**name**
The name of the certificate repository that the NSSD was unable to open.

System action
Processing continues.

Operator response
None.
System programmer response

Ensure that the repository name is defined correctly and that the user under which the NSSD was started is authorized to access the repository. When configured without the IBM Configuration Assistant for z/OS Communications Server, the certificate repository name is set on the KeyRing parameter of the NssConfig statement. See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference. When configured with the IBM Configuration Assistant for z/OS Communications Server, the certificate repository name is set in the key ring database name located in the Image Information: NSSD Settings panel.

User response

None.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP: Network Security Server

Module

CertRepository.cpp

Routing code

10

Descriptor code

12

Example

EZD1330I ERROR ( 3353006 | FILE OR KEYRING NOT FOUND ) WHILE OPENING CERTIFICATE REPOSITORY

EZD1331I name IS NOT A VALID CERTIFICATE REPOSITORY

Explanation

The repository specified on the NssConfig statement is not a certificate repository.

In the message text:

name

The repository specified on the KeyRing parameter of the NssConfig statement.

System action

Processing continues.

Operator response

Contact the system programmer.
**System programmer response**

Specify a valid repository name on the KeyRing parameter of the NssConfig statement in the network security services (NSS) server configuration file. See the information about the NssConfig statement in z/OS Communications Server: IP Configuration Reference for more information.

**User response**

None.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: Network Security Server

**Module**

CertRepository.cpp

**Routing code**

10

**Descriptor code**

12

**Example**

```
EZD1331I BADKEYRING IS NOT A VALID CERTIFICATE REPOSITORY
```

**Explanation**

A syntax error was detected in the specified parmlib member, which is used to configure network security services daemon (NSSD) CTRACE options. NSSD CTRACE is initialized with the minimum tracing option.

In the message text:

`pname`

The name of the CTRACE parmlib member.

**System action**

NSSD CTRACE initializes with the minimum tracing option; NSSD continues.

**Operator response**

Correct the syntax error in the parmlib member. See the information about the Sockets API traces in z/OS Communications Server: IP Diagnosis Guide for more information about configuring the CTrace parmlib member.

**System programmer response**

None.

**User response**

Not applicable.
Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssTrace.cpp

Routing code
10

Descriptor code
12

Example
EZD1332I SYNTAX ERROR IN NSSD CTRACE PARMLIB MEMBER CTINSS00

EZD1333I KEYWORD OR KEYVALUE REJECTED - KEYWORD keyword KEYVALUE keyvalue

Explanation
The network security services daemon (NSSD) was processing a configuration file and either the keyword is unsupported or the key value is not valid for the keyword.

In the message text:

keyword
The keyword portion of the configuration line.

keyvalue
The value entered for the keyword.

System action
The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes will be committed and the server continues using the old configuration values. If this error occurs during the initial startup sequence of the server, then the server will not start.

Operator response
Contact the system programmer.

System programmer response
See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Verify that the configuration file contains valid keywords and key values.

User response
No action needed.

Problem determination
None.
**Example**

| EZD1333I KEYWORD OR KEYVALUE REJECTED - KEYWORD PORTTT KEYVALUE 4900 |
|-------------------------|-------------------------|
| EZD1333I KEYWORD OR KEYVALUE REJECTED - KEYWORD PORT KEYVALUE -4900 |

**EZD1334I**    RIGHT BRACE (}) EXPECTED, BUT NOT FOUND

**Explanation**

The network security services daemon (NSSD) was processing a configuration file and the right brace (}) of a configuration statement was missing. All statements begin with a left brace ({) and end with a right brace (}). Each brace must be on a separate line.

**System action**

The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues using the old configuration values. If this error occurs during the initial startup sequence of the server, then the server does not start.

**Operator response**

Contact the system programmer.

**System programmer response**

See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Verify that the configuration file contains valid keywords and key values.

**User response**

No action needed.

**Problem determination**

None

**Source**

z/OS Communications Server TCP/IP: Network Security Server

**Module**

StatementParser.cpp
Routing code
10

Descriptor code
12

Example
Not applicable.

EZD1335I  statementname IS NOT A RECOGNIZED STATEMENT TYPE

Explanation
An unrecognized statement type occurs in the network security services daemon (NSSD) configuration file.
In the message text:

statementname
The name of the statement that appears on the line preceding the left brace ({}).

System action
The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues using the old configuration values. If this error occurs during the initial startup sequence of the server, then the server does not start.

Operator response
Contact the system programmer.

System programmer response
If this message was unexpected, see the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Verify that the configuration file contains valid contents.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
ConfigFileParser.cpp

Routing code
10

Descriptor code
12
Example
If the NssConfig statement name is spelled incorrectly, as in the following example, then the parser stops processing the configuration file:

```
NssConfigggg
{
  PORT 4900
}
```

EZD1335I NssConfigggg IS NOT A RECOGNIZED STATEMENT TYPE

EZD1336I THE CONFIGURATION VALUE value ENTERED FOR keywordname IS OUTSIDE THE ALLOWABLE RANGE lowvalue - highvalue

Explanation
The value configured for the specified network security services daemon (NSSD) keyword is outside the allowable range.

In the message text:

- **value**: The value configured for the keyword in the NSSD configuration file.
- **keywordname**: The name of the NSSD configuration keyword.
- **lowvalue**: The lowest value allowed for the keyword.
- **highvalue**: The highest value allowed for the keyword.

System action
The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues using the old configuration values. If this error occurs during the initial startup sequence of the server, then the server does not start.

Operator response
Contact the system programmer.

System programmer response
See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Correct the error in the configuration file and restart NSSD or reissue the MODIFY command.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssStatementParser.cpp
Routing code
10

Descriptor code
12

Automation
This message is written to both the syslog and the system console.

Example

```
EZD1336I THE CONFIGURATION VALUE 9999 ENTERED FOR SYSLOGLEVEL IS OUTSIDE THE ALLOWABLE RANGE 0 - 255
```

```
EZD1337I NSS SERVER IS USING TCP PORT portvalue
```

Explanation
The TCP port of the network security services (NSS) server was successfully initialized or changed during the processing of a configuration file. This message is logged the first time the server starts up and each time thereafter that a MODIFY NSSD,REFRESH command causes the server to change to a different port.

In the message text:

`portvalue`

The TCP port that the NSS server will be listening to.

System action
Processing continues. Any existing NSS client connections are maintained on the previously configured port.

Operator response
No action needed.

System programmer response
No action needed.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssReadConfiguration.cpp

Routing code
10
EZD1338I   NSS SERVER IS USING THE KEY RING keyringvalue

Explanation
The network security services (NSS) server key ring was successfully initialized or changed during the processing of the configuration file. This message is logged the first time the server starts up and each time thereafter that a REFRESH command causes the server to change to a different key ring.

In the message text:

keyringvalue
   The name of the key ring that the NSS server is using for creating and verifying signatures.

System action
Processing continues.

Operator response
No action needed.

System programmer response
No action needed.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssReadConfiguration.cpp

Routing code
10

Descriptor code
12

Example
EZD1338I NSS IS USING THE KEY RING NSSD2/KEYRING
THE LENGTH \textit{partlength} OF THE \textit{parttype} PORTION OF THE KEYRING PARAMETER \textit{keyringvalue} IS OUTSIDE OF THE ACCEPTABLE RANGE OF \textit{lowvalue} - \textit{highvalue}

\textbf{Explanation}

The network security services daemon (NSSD) was processing a configuration file and either the user ID portion or the ring name portion of the Keyring parameter is the wrong length.

In the message text:

\textit{partlength}
\begin{itemize}
  \item The length of the value as it appeared in the configuration file.
\end{itemize}

\textit{parttype}
\begin{itemize}
  \item Possible values are \textit{userid} or \textit{keyring}.
\end{itemize}

\textit{keyringvalue}
\begin{itemize}
  \item The value entered for the keyword.
\end{itemize}

\textit{lowvalue}
\begin{itemize}
  \item The shortest possible length of the value.
\end{itemize}

\textit{highvalue}
\begin{itemize}
  \item The longest possible length of the value.
\end{itemize}

\textbf{System action}

The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues using the old configuration values. If this error occurs during the initial startup sequence of the server, then the server does not start.

\textbf{Operator response}

See the system programmer.

\textbf{System programmer response}

See the information about the network security services (NSS) server in \textit{z/OS Communications Server: IP Configuration Reference} for information about the NSSD configuration file. Verify that the configuration file contents are valid.

\textbf{User response}

No action needed.

\textbf{Problem determination}

None.

\textbf{Source}

\textit{z/OS Communications Server TCP/IP: Network Security Server}

\textbf{Module}

\textit{NssStatementParser.cpp}

\textbf{Routing code}

10

\textbf{Descriptor code}

12
Automation

This message is written to both the syslog and the system console.

Example

EZD1339I THE LENGTH 12 OF THE USERID PORTION OF THE KEYRING PARAMETER USER00001234/KEYRING IS OUTSIDE OF THE ACCEPTABLE RANGE OF 1 - 8
EZD1339I THE LENGTH 0 OF THE RINGNAME PORTION OF THE KEYRING PARAMETER USER023/ IS OUTSIDE OF THE ACCEPTABLE RANGE OF 1 - 237

EZD1340I THE keyword VALUE keyvalue CONTAINS CHARACTERS THAT ARE NOT 0-9

Explanation

The network security services daemon (NSSD) configuration file contains a value that is not a valid decimal number. The value must contain only the integers in the range 0–9. Negative numbers are not allowed.

In the message text:

keyword
  The keyword portion of the configuration line.

keyvalue
  The value entered for the keyword.

System action

The processing of the configuration file stops. If this occurs during a MODIFY command, then no changes are committed and the server continues using the old configuration values. If this occurs during the initial startup sequence of the server, then the server does not start.

Operator response

Contact the system programmer.

System programmer response

See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Verify that the configuration file contents are valid.

User response

No action needed.

Problem determination

None.

Source

z/OS Communications Server TCP/IP: Network Security Server

Module

NssStatementParser.cpp

Routing code

10
Descriptor code
12

Automation
This message is written to both the syslog and the system console.

Example

```
EZD1340I THE PORT VALUE -4999 MAY CONTAIN ONLY THE DECIMAL CHARACTERS 0-9
EZD1340I THE PORT VALUE STARBOARD MAY CONTAIN ONLY THE DECIMAL CHARACTERS 0-9
```

EZD1341I THE NSS SERVER PORT VALUE portvalue IS INCORRECT BECAUSE IT IS OUTSIDE OF THE ALLOWABLE RANGE 1-maxport

Explanation
In the network security services daemon (NSSD) configuration file, the port value is not in the range supported by the server.

In the message text:

- **portvalue**
  - The value specified for the port.

- **maxport**
  - The maximum value of the port range portvalue is the value that was specified.

System action
The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues using the old configuration values. If this error occurs during the initial startup sequence of the server, then the server does not start.

Operator response
See the system programmer.

System programmer response
No action needed.

User response
See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Verify that the configuration file contains valid contents.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssStatementParser
Routing code
10

Descriptor code
12

Automation
This message is written to both the syslog and the system console.

Example

```
EZD1341I THE NETWORK SECURITY SERVER PORT VALUE 750000 IS INCORRECT BECAUSE IT IS OUTSIDE OF THE ALLOWABLE RANGE 1-65535
```

```
EZD1342I THE NSS SERVER CANNOT PROVIDE CERTIFICATE SERVICES USING THE CURRENTLY CONFIGURED CERTIFICATE REPOSITORY NAME (repositoryname)
```

Explanation
The Keyring parameter of the network security services daemon (NSSD) does not specify a usable certificate repository. The Keyring value might appear to be syntactically valid, but it is not correctly configured for use by the user ID under which NSSD is running. Certificate related services cannot be provided.

In the message text:

`repositoryname`
   The name of the certificate repository as configured with the Keyring parameter. This field will be blank if no key ring was specified in the configuration file.

System action
NSSD continues without support for remote certificate related services.

Operator response
Contact the system programmer.

System programmer response
See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information. Correct the error in the configuration file and restart NSSD or reissue the MODIFY command.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssReadConfiguration.cpp
Routing code
10

Descriptor code
12

Example

EZD1342I THE NSS SERVER CANNOT PROVIDE CERTIFICATE SERVICES USING THE CURRENTLY CONFIGURED CERTIFICATE REPOSITORY NAME (USER100/MYKEYRING)

**EZD1343I THE VALUE IS MISSING FOR NETWORK SECURITY SERVER CONFIGURATION KEYWORD**

*keywordname*

**Explanation**
The key word requires a value but no value was specified in the network security services (NSS) server configuration file.

In the message text:

*keywordname*  
The name of the keyword that is missing a value.

**System action**
The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues using the previous configuration values. If this error occurs during the initial startup sequence of the server, then the server does not start.

**Operator response**
Contact the system programmer.

**System programmer response**
See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Correct the error in the configuration file and restart NSSD or reissue the MODIFY command.

**User response**
No action needed.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssStatementParser.cpp

**Routing code**
10
Descriptor code
12

Automation
This message is written to both the syslog and the system console.

Example

EZD1343I THE VALUE IS MISSING FOR NETWORK SECURITY SERVER CONFIGURATION KEYWORD SYSLOGLEVEL

EZD1344I AN ERROR OCCURRED WHILE READING THE NSS SERVER CONFIGURATION FILE configfile - RETURN CODE retcode

Explanation
A general error occurred while processing the network security services daemon (NSSD) configuration file. Additional messages will be issued to provide more specific information.

In the message text:

configfile
The name of the configuration file that is being processed.

retcode
The general configuration parsing return code. Possible values are:

2
Cannot open the configuration file.

3
There was a syntax or value error in the file.

System action
The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues using the old configuration values. If this error occurs during the initial startup sequence of the server, then the server does not start.

Operator response
Contact the system programmer.

System programmer response
See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Verify that the configuration file is present and that the server has permission to read it.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server
Module
NssMainThread.cpp

Routing code
10

Descriptor code
12

Example

EZD1344I AN ERROR OCCURRED WHILE READING THE NSS SERVER CONFIGURATION FILE, /etc/badpath/nssd.conf - RETURN CODE 2

EZD1345I THIS INSTANCE OF THE NSS SERVER CANNOT START BECAUSE THERE IS ALREADY ANOTHER INSTANCE OF THE SERVER RUNNING ON THIS SYSTEM - TOKEN tokenname LEVEL level PERSIST persist RETURN CODE retcode

Explanation

Only one network security services daemon (NSSD) can be running at the same time on the same system.

In the message text:

tokenname
The name of the MVS token that the server is trying to get exclusive access to. This value is NSSD for the NSSD.

level
The level of exclusivity required. This value is 4 for NSSD.

persist
Possible values are:

• 0 if the token is released when the server ends
• 1 if it persists after the server ends.

This value is 0 for NSSD.

retcode
The return code from the MVS token service. See the z/OS MVS Programming: Authorized Assembler Services Reference EDT-IXG for a complete list of IEANTCR return and reason codes.

System action

This instance of the network security services (NSS) server does not start.

Operator response

If a copy of the NSSD is already running, then it must be shut down before starting a new instance. If the NSSD is not running, contact the system programmer.

System programmer response

Contact the IBM Service Center with this message.

User response

No action needed.
Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp

Routing code
10

Descriptor code
12

Example
EZD1345I THIS INSTANCE OF THE NSS SERVER CANNOT START BECAUSE THERE IS ALREADY ANOTHER
INSTANCE OF THE SERVER RUNNING ON THIS SYSTEM - TOKEN NSSD LEVEL 4 PERSIST 0 RETURN
CODE 4

EZD1346I LEFT BRACE ( { ) EXPECTED, BUT NOT FOUND.

Explanation
The left brace ( {) of a configuration statement was missing. All statements begin with a left brace ( {) and end
with a right brace (}). Each brace must be on a separate line.

System action
The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes
are committed and the server continues using the old configuration values. If this error occurs during the initial
startup sequence of the server, then the server does not start.

Operator response
See the system programmer.

System programmer response
See the information about the network security services (NSS) server in z/OS Communications Server: IP
Configuration Reference for information about the NSSD configuration file. Verify that the configuration file
contents are valid. In particular, look for unmatched braces.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server
Module
StatementParser.cpp

Routing code
10

Descriptor code
12

Example
Not applicable.

**EZD1347I** INTERNAL ERROR *errid* IN MODULE *modname* - UNABLE TO OBTAIN MEMORY OF SIZE *size*

**Explanation**
The network security services daemon (NSSD) was unable to obtain the required amount of memory.

In the message text:

*errid*
The code that helps IBM service representatives identify the specific storage allocation request.

*modname*
The name of the module that encountered the error.

*size*
The amount of storage requested, in bytes.

**System action**
The current operation fails and the NSSD attempts to continue.

**Operator response**
Ensure that there is enough memory available on the system and try the operation again. The module name and error ID can be used to determine the exact location of the storage allocation request, if that level of problem determination is required. If so, contact the system programmer with these values.

**System programmer response**
If you contact IBM software support services in the process of correcting this problem, provide the module name *modname* and error ID *errid* so that the IBM software support services representative can identify the specific storage allocation call that failed.

**User response**
None.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

Module
Numerous
**Routing code**
10

**Descriptor code**
12

**Example**

```
EZD1347I INTERNAL ERROR 9 IN MODULE EZAINCLS - UNABLE TO OBTAIN MEMORY OF SIZE 840154
```

```
EZD1348I THE MODIFY COMMAND EXCEEDED THE MAXIMUM ALLOWED LENGTH OF maxlen
```

**Explanation**
The network security services (NSS) server MODIFY command can accept only the first `maxlen` value entered.

In the message text:

`maxlen`  
The maximum number of characters that the MODIFY command can accept.

**System action**
Processing continues using the previous configuration values.

**Operator response**
Reissue the MODIFY command without exceeding the length limit.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssModifyComandHandler.cpp

**Routing code**
10

**Descriptor code**
12

**Example**

```
EZD1348I THE MODIFY COMMAND EXCEEDED THE MAXIMUM ALLOWED LENGTH OF 128
```
EZD1349I  THE COMMAND ENTERED IS NOT A RECOGNIZED MODIFY REQUEST - input

Explanation
The operator entered an unsupported MODIFY request.
In the message text:

input
   The MODIFY command text entered by the operator.

System action
The command is ignored.

Operator response
Correct and reissue the command.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssModifyCommandHandler.cpp

Routing code
10

Descriptor code
12

Example
EZD1349I  THE COMMAND ENTERED IS NOT A RECOGNIZED MODIFY REQUEST - zzzzz

EZD1350I  THE MODIFY SUBCOMMAND IS NOT SUPPORTED BY THE NSS SERVER - subcommand

Explanation
The operator has entered a MODIFY subcommand that is not supported by the network security services (NSS) server.
In the message text:

subcommand
   The portion of the MODIFY command that the operator entered on the console.
See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for more information about the MODIFY command.

**System action**
The command is ignored.

**Operator response**
Correct and reissue the command.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssModifyCommandHandler.cpp

**Routing code**
10

**Descriptor code**
12

**Example**
If the operator enters `MODIFY nssd,Hello World` then the following message would be sent to the console:

```
EZD1350I THE MODIFY COMMAND OPTION IS NOT SUPPORTED BY THE NSS SERVER - HELLO WORLD
```

**Explanation**
The network security services (NSS) server has begun its shutdown sequence.

**System action**
The server is in the process of shutting down. No more work is processed.

**Operator response**
None.
System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp

Routing code
10

Descriptor code
12

Example
Not applicable.

**EZD1352I**  NSS SERVER RECEIVED THE STOP COMMAND

Explanation
The network security services (NSS) server recognizes that a STOP command was issued on the console.

System action
The NSS Server begins its shutdown sequence.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server
Module
NssModifyCommandHandler.cpp

Routing code
10

Descriptor code
12

Example

EZD1352I NSS SERVER RECEIVED THE STOP COMMAND

EZD1353I NSS SERVER CONFIG PROCESSING COMPLETE USING FILE filename

Explanation
The network security services (NSS) server successfully completed processing the configuration file. This message can follow the successful startup of the server or a successful MODIFY REFRESH command.

In the message text:

filename
The US file or MVS data set name that contains NSS server configuration statements.
- USS file names are delimited by single quotation marks ('), as in this example: '/etc/security/nssd.conf'
- MVS data set names are prefixed with double slashes ('//') and the remainder of the data set name is enclosed in single quotation marks ('), as in this example: '//MVS.FILENAME(MEMBER)'

System action
None.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssModifyCommandHandler.cpp
Routing code
10

Descriptor code
12

Example

EZD1353I NSS SERVER CONFIG PROCESSING COMPLETE USING FILE /etc/security/nssd.conf

EZD1354I NSS SERVER ERROR rc OCCURRED WHILE PROCESSING filename - MODIFY REFRESH COMMAND IS REJECTED

Explanation
An error occurred during the processing of a configuration file in response to a MODIFY REFRESH command. In the message text:

rc
- The return code. Possible values are:
  2 The file does not exist or could not be opened.
  3 An error was detected while processing one of the statements in the file.

filename
- The name of the configuration file that was being processed.

System action
The processing of the configuration file stops. No changes are committed and the server continues using the old configuration values.

Operator response
Reissue the MODIFY command using the correct file name. If problems persist, contact the system programmer.

System programmer response
See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Verify that the configuration file exists, that it is accessible to the server and that the file content is valid.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssModifyCommandHandler.cpp
Routing code
10

Descriptor code
12

Example

EZD1354I NSS SERVER ERROR 2 OCCURRED WHILE PROCESSING /etc/security/nssd.conf3 - MODIFY REFRESH COMMAND IS REJECTED

EZD1355I INCORRECT VALUE FOR THE FILE KEYWORD ON THE NSS SERVER MODIFY COMMAND - *file*

Explanation
The file name was specified incorrectly on a MODIFY procname,REFRESH,FILE=*file* command.
In the message text:

*file*
   The incorrect file name.

System action
The MODIFY command is ignored. Processing continues using the previous configuration values.

Operator response
Correct the file name and try again. If problems persist, contact the system programmer.

System programmer response
See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssModifyCommandHandler.cpp

Routing code
10

Descriptor code
12
Example

EZD1355I INCORRECT VALUE FOR THE FILE KEYWORD ON THE NSS SERVER MODIFY COMMAND - ////nssd.conf

EZD1356I NSS SERVER SHUTDOWN SEQUENCE HAS COMPLETED

Explanation
The network security services (NSS) server has completed its shutdown sequence and is terminating.

System action
The NSS Server ends.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp

Routing code
10

Descriptor code
12

Example

EZD1356I NSS SERVER SHUTDOWN SEQUENCE HAS COMPLETED

EZD1357I NSS SERVER INITIALIZATION SEQUENCE HAS BEGUN

Explanation
This is a notification that the server has begun its initialization sequence.

System action
The network security services (NSS) server is starting up.
Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp

Routing code
10

Descriptor code
12

Example
EZD1357I NSS SERVER INITIALIZATION SEQUENCE HAS BEGUN
EZD1358I NSS SERVER INITIALIZATION SEQUENCE HAS COMPLETED

Explanation
This is a notification that the network security services (NSS) server has completed its initialization sequence.

System action
The NSS Server is ready.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.
Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp

Routing code
10

Descriptor code
12

Example
EZD1358I NSS SERVER INITIALIZATION SEQUENCE HAS COMPLETED

EZD1359I NSS SERVER RELEASE release SERVICE LEVEL level CREATED ON date

Explanation
This message is the first message printed to the console when the network security services (NSS) server is started.

In the message text:

release
The release name.

level
The service level name.

date
The server build date.

System action
None.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server
Module
NssMainThread.cpp

Routing code
10

Descriptor code
12

Example
EZD1359I NSS SERVER RELEASE CS V1R9 SERVICE LEVEL CS060716 CREATED ON Jul 16 2006

EZD1360I  TCP PORT portnumber IS CURRENTLY UNAVAILABLE TO THE NSS SERVER

Explanation
The network security services (NSS) server cannot connect to the specified port because the port is unavailable. The port might be in use by another program, it might be reserved for another program or NSS might not have the authority to access the port.

In the message text:

portnumber  The TCP port to which the NSS server is trying to bind.

System action
If this message is issued during server startup, then the server does not start. If this message is issued during a MODIFY REFRESH command, the refresh is rejected and the server continues to run using the configuration values that were in place prior to the refresh attempt.

Operator response
Contact the system programmer.

System programmer response
The network security services (NSS) server must be running on a unique TCP port. In the case of a conflict, either the NSS server or the conflicting program must be reconfigured to use a different TCP port. If NSS does not have access to the port, verify that the port is not reserved for another program and that NSS has been granted sufficient authority to access the port. See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp
Routing code
10

Descriptor code
12

Example
EZD1360I TCP PORT 4503 IS CURRENTLY UNAVAILABLE TO THE NSS SERVER

EZD1361A NSS SERVER IS WAITING FOR A TCP/IP STACK TO START

Explanation
The network security services (NSS) server is waiting for a stack to start.

System action
The NSS server suspends processing until a TCP/IP stack is available.

Operator response
Ensure that a TCP/IP stack is up and running on the NSS server machine. See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for more information.

System programmer response
No action needed.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp

Routing code
10

Descriptor code
12

Example
EZD1361A NSS SERVER IS WAITING FOR A TCP/IP STACK TO START

EZD1362A NSS SERVER IS WAITING FOR TCP PORT portnumber
Explanation
The network security services (NSS) server has been configured to listen on TCP the specified port but it cannot bind to that port because another process is currently using it or because the port is not configured for access.

In the message text:

**portnumber**
The TCP port number to which the NSS server is trying to bind.

System action
The NSS server continues to wait for its port to become available. This message will remain on the console until one of the following occurs:

- The port is released and the NSS server is able to bind to it. Processing continues normally from this point.
- An unrecoverable error occurred while attempting to bind to the port. The NSS server ends.
- The operator clears the message from the console manually. The NSS server continues to wait for its port. Processing continues when either the port is obtained or the NSS server ends.

Operator response
Contact the system programmer.

System programmer response
Verify that the NSS server is configured to use the correct TCP port. If the NSS server is configured correctly, then determine which other process is using the TCP port or whether there is a configuration error for the TCP port itself. See z/OS Communications Server: IP Configuration Guide for information about configuring the NSS server and z/OS TCP ports.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
ServerILM.cpp

Routing code
10

Descriptor code
12

Example

<table>
<thead>
<tr>
<th>EZD1362A</th>
<th>NSS SERVER IS WAITING FOR TCP PORT 4519</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1363I</td>
<td>EXCEPTION classname ENCOUNTERED IN MODULE modname - ERROR ID internal_error_ID rc ERRNO_errno ERRNOJR_errnojr</td>
</tr>
</tbody>
</table>
Explanation
A runtime exception was issued.

In the message text:

classname
  The name of the exception class.

modname
  The name of the module in which the exception was detected.

internal_error_ID
  An internal error ID used by IBM to identify the code that detected the error

rc
  The return code, if any, from the function call that caused or detected the error condition.

errno
  The runtime errno value, if any, at the time the exception was detected.

errnojr
  The runtime errno2 value, if any, at the time the exception was detected.

System action
Processing continues in most cases. In some cases the network security services (NSS) server might experience problems or end.

Operator response
Contact the system programmer

System programmer response
This message is not always a sign of an unrecoverable network security services daemon (NSSD) condition. However, any traces or logs containing this message should be forwarded to IBM Service if the server is unable to operate normally after these messages have been logged.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
Exception.cpp

Routing code
10

Descriptor code
12
Example

EZD1363I EXCEPTION CONSTRUCTOR ERROR ENCOUNTERED IN MODULE EZAINCEM - ERROR ID 1 RC 1 ERRNO 0 ERRNOJR 0

EZD1364I NSS SERVER CANNOT WRITE PROCESS ID NUMBER pid TO filename - ERRNO errno ERRNO description

Explanation
A file system error occurred when the network security services (NSS) server tried to write its process ID number to a file.

In the message text:

pid
The NSS server process ID.

filename
The file into which the NSS server was trying to write its process ID number.

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

description
Describes the meaning of the errno value.

System action
Network security services daemon (NSSD) processing continues.

Operator response
None.

System programmer response
This is not a problem unless the pid file will be used for automation (for example, to automate ending the NSSD).
Information about setting the pid file location can be found in the z/OS Communications Server: IP Configuration Guide.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp

Routing code
10
EZD1364I NSS SERVER CANNOT WRITE PROCESS ID NUMBER 83886209 TO /tmp-baddir/nssd.pid - ERRNO 129 ERRNO DESCRIPTION EDC5129I NO SUCH FILE OR DIRECTORY.

EZD1365I A MESSAGE-GENERATED DUMP HAS BEEN CREATED WITH TITLE title

Explanation
A network security services (NSS) server syslog message generated an address space dump. The message that generated the dump appears immediately after this message in the system log.

In the message text:

<table>
<thead>
<tr>
<th>title</th>
</tr>
</thead>
<tbody>
<tr>
<td>The text associated with the dump. The title contains the message number and associated message text that caused the dump to be generated.</td>
</tr>
</tbody>
</table>

System action
After the dump is created, the network security services daemon (NSSD) continues processing.

Operator response
Contact the system programmer.

System programmer response
Capture the system log and the generated dump. Contact IBM software support services to analyze this data.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssLog.cpp

Routing code
10

Descriptor code
12

Example
EZD1365I A MESSAGE-GENERATED DUMP HAS BEEN CREATED WITH TITLE NSSD MESSAGE GENERATED dump EZD1322I INTERNAL ERROR IN MODULE EZAINMAI 0003 - 0 | 0 | 0
EZD1366I A MESSAGE-GENERATED DUMP WAS SUPPRESSED FOR MESSAGE message

Explanation
A network security services (NSS) server syslog message attempted to generate an address space dump. However, no more than two message-generated dumps can be created in a 15-minute period, so the dump was suppressed. The message that attempted to generate the dump appears immediately after this message in the system log.

In the message text:
message
The message that attempted to generate the dump.

System action
Network security services daemon (NSSD) processing continues.

Operator response
Contact the system programmer.

System programmer response
Capture the system log and any message-generated dumps that were created earlier. Contact IBM software support services to analyze this data.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssLog.cpp

Routing code
10

Descriptor code
12

Example

EZD1366I A MESSAGE-GENERATED DUMP WAS SUPPRESSED FOR MESSAGE message

EZD1322I INTERNAL ERROR IN MODULE EZAINMAI 0003 - 0 | 0 | 0

EZD1367I NSS discipline_type CLIENT client_name CONNECTED TO THE NSS SERVER USING NULL-ENCRYPTION
**Explanation**
The network security services (NSS) server and the NSS client negotiated a connection that used the NULL encryption algorithm, but the connection should be encrypted.

In the message text:

- **discipline_type**
  The discipline that the NSS client is using.

- **client_name**
  The name of the client that is connected to the NSS server.

**System action**
The NSS server continues to process requests over the unencrypted session.

**Operator response**
Contact the system programmer.

**System programmer response**
Modify the AT-TLS policy to require encryption of connections to the NSS server. See the information about the AT-TLS policy statements in z/OS Communications Server: IP Configuration Reference for more information about AT-TLS policy statements.

**User response**
No action needed.

**Problem determination**
None

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssClient.cpp

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to both the operator console and syslog.

**Example**

```
EZD1367I NSS XMLAPPLIANCE CLIENT XMLAPPLIANCEDOMAIN1 CONNECTED TO THE NSS SERVER USING NULL-ENCRYPTION

EZD1368I NSS SERVER CONFIGURATION FILE parm PARAMETER DOES NOT SUPPORT VALUE - value
```
Explanation
An unsupported value was configured for one of the network security services (NSS) parameters.

In the message text:

parm
The parameter that contains the unsupported value.

value
The unsupported value.

System action
Processing of the configuration file stops. If this error occurs during the processing of a MODIFY command, no changes are committed and the server continues using the old configuration values. If this error occurs during the server startup sequence, the server does not start.

Operator response
Contact the system programmer.

System programmer response
Verify that the configuration file contains valid keywords and key values. See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file.

User response
No action needed.

Problem determination
None

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssStatementParser.cpp

Routing code
10

Descriptor code
12

Automation
This message is written to both the operator console and syslog.

Example

<table>
<thead>
<tr>
<th>Code</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1368I</td>
<td>NSS SERVER CONFIGURATION FILE DISCIPLINE PARAMETER DOES NOT SUPPORT VALUE - IPSEC</td>
</tr>
<tr>
<td>EZD1369I</td>
<td>NSS SERVER CONFIGURATION FILE parm PARAMETER CONTAINS EXTRANEOUS INFORMATION - text</td>
</tr>
</tbody>
</table>

962  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**Explanation**
Extraneous information was configured for one of the network security services (NSS) parameters.

In the message text:

*parm*  
The parameter that contains extraneous information.

*text*  
The extraneous information.

**System action**
Configuration file processing stops. If this error occurs during a MODIFY command, no changes are committed and the server continues using the old configuration values. If this error occurs during the server startup sequence, the server does not start.

**Operator response**
Contact the system programmer.

**System programmer response**
Verify that the configuration file contains valid keywords and key values. See the information about the network security services (NSS) server in *z/OS Communications Server: IP Configuration Reference* for information about the NSSD configuration file.

**User response**
No action needed.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssStatementParser.cpp

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to both the operator console and syslog.
Example
If the system programmer configured Discipline IPSec Enable Discipline XMLAppliance Disable in the NSS configuration file, the following message would be issued:

```
EZD1369I NSS SERVER CONFIGURATION FILE DISCIPLINE PARAMETER CONTAINS EXTRANEOUS INFORMATION - DISCIPLINE XMLAPPLIANCE DISABLE
```

**EZD1370I INCORRECT SYNTAX ON THE MODIFY NSS SERVER SUBCOMMAND subcommand**

**Explanation**
The syntax of the MODIFY network security services (NSS) server command is incorrect. The syntax error occurs after the subcommand.

In the message text:

**subcommand**
The subcommand portion of the MODIFY command that the operator entered on the console. The syntax error occurs after the subcommand.

**System action**
The command is ignored.

**Operator response**
Correct and reissue the command. See the information about the NSS MODIFY command in z/OS Communications Server: IP System Administrator's Commands for the command syntax.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssModifyCommandHandler.cpp

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is sent to the console.
Example
If the operator issued the MODIFY NSSD,DISPLAY,ALL command, the following message is issued:

EZD1370I INCORRECT SYNTAX ON THE MODIFY NSS SERVER SUBCOMMAND DISPLAY

EZD1371I AN NSS CLIENT ATTEMPTED TO USE A DISABLED DISCIPLINE discipline_type TO CONNECT TO THE NSS SERVER - THE CONNECTION IS CLOSED

Explanation
A network security services (NSS) client attempted to use a disabled discipline to connect to the NSS server. The connection is closed.

In the message text:

 discipline_type
   The discipline that is disabled.

System action
The NSS server closes the NSS client TCP connection.

Operator response
Contact the system programmer.

System programmer response
Modify the NSS configuration file to enable the specified discipline. See the information about the network security services (NSS) server in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
ConnectClientReqToSrv_server.cpp

Routing code
10

Descriptor code
12

Automation
This message will be written to the syslog.
EZD1372I MODIFY NSS SERVER subcommand SUBCOMMAND ACCEPTED

Explanation
The network security services (NSS) server accepted the MODIFY subcommand.

In the message text:

subcommand
The MODIFY subcommand that was accepted.

System action
The NSS server continues processing the MODIFY command.

Operator response
None.

System programmer response
None.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssModifyCommandHandler.cpp

Routing code
10

Descriptor code
12

Automation
This message is written to both the operator console and syslog.

Example

EZD1372I MODIFY NSS SERVER DISPLAY SUBCOMMAND ACCEPTED

EZD1373I NSS discipline_name DISCIPLINE state
Explanation
This message is displayed during initialization of the network security services (NSS) server and whenever the state of one of the configured disciplines changes.

In the message text:

discipline_name
The discipline state that is being displayed.

state
The state of the discipline. Possible values are ENABLED or DISABLED.

System action
The NSS server continues processing requests for enabled disciplines. When a discipline is disabled, any connections using that discipline are disconnected. New NSS clients that request a disabled discipline are not allowed to connect.

Operator response
None.

System programmer response
None.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssReadConfiguration.cpp

Routing code
10

Descriptor code
12

Automation
This message is written to both the operator console and syslog.

Example

<table>
<thead>
<tr>
<th>EZD1373I NSS IPSEC DISCIPLINE ENABLED</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1373I NSS XMLAPPLIANCE DISCIPLINE ENABLED</td>
</tr>
</tbody>
</table>

EZD1374E   ICSF services are currently unavailable to the NSS daemon
**Explanation**
The network security services (NSS) server detected that Integrated Cryptographic Services Facility (ICSF) services are unavailable. The NSS server uses ICSF services for various cryptographic functions. When ICSF services are not correctly configured and started, the NSS server is not fully functional to NSS clients. See the information about network security services in z/OS Communications Server: IP Configuration Guide for information about which NSS server functions require ICSF.

**System action**
The NSS daemon continues to run, but it cannot provide the services that rely on ICSF.

**Operator response**
Contact the system programmer.

**System programmer response**
Configure and start ICSF. See the z/OS Cryptographic Services ICSF Administrator's Guide for more information.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssAnchor.cpp

**Routing code**
3

**Descriptor code**
3

**Automation**
This message goes to the console and to syslog.

**Example**

<table>
<thead>
<tr>
<th>EZD1374E</th>
<th>ICSF services are currently unavailable to the NSS daemon</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1375I</td>
<td><strong>THE keyword PARAMETER VALUE value EXCEEDS THE MAXIMUM ALLOWABLE LENGTH OF maxlen</strong></td>
</tr>
</tbody>
</table>
**keyword**
The configuration parameter on which the erroneous value is specified.

**value**
The erroneous value.

**maxlength**
The maximum allowable length for this configuration value.

**System action**
Processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues using the old configuration values. If this error occurs during the initial startup sequence of the server, then the server exits.

**Operator response**
Contact the system programmer.

**System programmer response**
See the information about the network security services (NSS) server information in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. Correct the configuration file and then restart NSSD or reissue the MODIFY command.

**User response**
No action needed.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssStatementParser.cpp

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to both the syslog and the system console.

**Example**

EZD1375I THE CertificateURL PARAMETER VALUE certlabel012345678901234567890123 EXCEEDS THE MAXIMUM ALLOWABLE LENGTH OF 32 CHARACTERS

EZD1376I THE NSS SERVER IGNORED THE CHANGED CONFIGURATION FILE PARAMETER *pname* - VALUE REMAINS *current_value*
Explanation
The specified network security services (NSS) server configuration file parameter cannot be changed by using the MODIFY jobname,REFRESH command. The NSS server continues to use the specified value.

In the message text:

**pname**
The parameter that cannot be changed on a refresh.

**current_value**
The value that NSS continues to use.

System action
The NSS server ignores the change to this parameter and configuration processing continues.

Operator response
If you want to use the changed configuration file parameter for the NSS server, contact the system programmer.

System programmer response
If you want to use the changed configuration file parameter for the NSS server, stop and restart the server.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssConfiguration.cpp

Routing code
10

Descriptor code
12

Automation
Message EZD1376I is written to the system console and to syslogd.

Example

<table>
<thead>
<tr>
<th>EZD1376I</th>
<th>THE NSS SERVER IGNORED THE CHANGED CONFIGURATION FILE PARAMETER</th>
<th>FIPS140 - VALUE REMAINS</th>
</tr>
</thead>
</table>

| EZD1377I | FIPS 140 support is enabled for the IPSec Discipline of the NSS server |

970  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Explanation
This message reports that Federal Information Processing Standard (FIPS) publication 140 (FIPS 140) support is enabled for the IPSec Discipline of network security services (NSS) server. Cryptographic operations are performed by cryptographic modules that are designed to follow the Level 1 security requirements of FIPS 140.

System action
The NSS server processing continues.

Operator response
None.

System programmer response
If FIPS 140 support is not required for the NSS server, stop the server, configure FIPS140 No in the NSS server configuration file, and restart the server.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssReadConfiguration.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1377I FIPS 140 support is enabled for the IPSec Discipline of NSS server

EZD1378I FIPS 140 support is not enabled for the IPSec Discipline of the NSS server

Explanation
This message reports that Federal Information Processing Standard (FIPS) publication 140 (FIPS 140) support is not enabled for the IPSec Discipline of network security services (NSS) server. Cryptographic operations for the IPSec discipline may be performed by cryptographic modules that are not designed to follow the Level 1 security requirements of FIPS 140.
System action
The NSS server processing continues.

Operator response
If FIPS 140 support is required for the NSS server, contact the system programmer.

System programmer response
If FIPS 140 support is required for the NSS server, stop the server, configure FIPS140 Yes in the NSS server configuration file, and restart the server.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssReadConfiguration.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example

EZD1378I FIPS 140 support is not enabled for the IPSec Discipline of NSS server

EZD1379I FIPS140 Yes is rejected by System SSL code description

Explanation
The network security services (NSS) server initial configuration specified FIPS140 Yes but System SSL rejected the request.

In the message text:

code
The hexadecimal gsk_status code returned from gsk_fips_state_set().

description
The description of the code value provided by the System SSL.

System action
The NSS server ends.
Operator response
Contact the system programmer.

System programmer response
For more information about the error, see CMS status codes in z/OS Cryptographic Services System SSL Programming. If the error indicates a problem with software installation or system configuration, correct the error and restart the NSS server. If the error indicates a problem with the NSS server application, contact IBM support.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssReadConfiguration.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1379I FIPS140 Yes is rejected by System SSL 0335306C Attempt to execute in FIPS mode failed.

EZD1380I THE NSS SERVER IS SHUTTING DOWN DUE TO AN ERROR DURING INITIALIZATION ( errno | errojr | description )

Explanation
The network security services (NSS) server encountered an error during initialization. The server is shutting down.

In the message text:

errno
The UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

errojr
The hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.
**description**
Describes the meaning of the *errno* value.

**System action**
The network security services (NSS) server shuts down.

**Operator response**
Contact the system programmer.

**System programmer response**
Correct the error indicated by the *errno*, *errnojr*, or *description* values. This message might be generated during server initialization as the result of a shortage of available disk space in the /var directory.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IPSec

**Module**
NssMainThread.cpp

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to both the syslog and the system console.

**Example**

```
EZD1380I THE NSS SERVER IS SHUTTING DOWN DUE TO AN ERROR DURING INITIALIZATION ( 133 | 4010237980 | EDC5133I NO SPACE LEFT ON DEVICE. )
```

**EZD1381I**  THE *parmvalue* VALUE IS MISSING FOR CONFIGURATION KEYWORD *keyword*

**Explanation**
The network security services daemon (NSSD) was processing a configuration file and encountered a configuration parameter that is missing an expected value. This message only appears for configuration parameters that expect multiple values.

In the message text:

*parmvalue*
   The value that is missing.
**keyword**
The configuration parameter from which the expected value is missing.

**System action**
The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues to use the old configuration values. If this error occurs during the initial startup sequence of the server, then the server does not start.

**Operator response**
Contact the system programmer.

**System programmer response**
Correct the configuration file and then restart NSSD or reissue the MODIFY command. See the information about the network security services (NSS) server information in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file.

**User response**
No response.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssStatementParser.cpp

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to both the syslog and the system console.

**Example**

```plaintext
EZD1381I THE URLVALUE IS MISSING FOR CONFIGURATION KEYWORD CertificateBundleURL

EZD1382I THE keyword PARAMETER URL VALUE urlstring IS FORMED INCORRECTLY: explanation
```

**Explanation**
The network security services daemon (NSSD) was processing a configuration file and encountered an incorrectly formed URL value on the specified configuration parameter.

In the message text:
The configuration parameter that contained the incorrect URL value.

The incorrectly formed URL value.

a description of the specific reason for which the URL is considered to be incorrect.

The processing of the configuration file stops. If this error occurs during a MODIFY command, then no changes are committed and the server continues using the old configuration values. If this error occurs during the initial startup sequence of the server, then the server does not start.

Contact the system programmer.

Verify the exact syntax of the URL for HTTP resource and correct the URL in the NSSD configuration file. For information about the rules for forming HTTP URLs, see IETF RFC 3986 and RFC 2616. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs. See the information about the network security services (NSS) server information in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file. When corrections have been made, restart NSSD or reissue the MODIFY command.

No response.

None.

z/OS Communications Server TCP/IP: Network Security Server

NssStatementParser.cpp

10

12

This message is written to both the syslog and the system console.

EZD1383I  FIPS140 support is enabled for the NSS daemon but it has read access to CRYPTOZ resource FIPSEXEMPT.SYSTOK-SESSION-ONLY

**Explanation**

When the NSS daemon is configured for Federal Information Processing Standards (FIPS) 140 mode, the NSS daemon must have no access privileges (NONE) to the SAF resource FIPSEXEMPT.SYSTOK-SESSION-ONLY in the CRYPTOZ class. See the NssConfig statement in z/OS Communications Server: IP Configuration Reference for information about the FIPS140 parameter.

**System action**

The NSS daemon ends.

**Operator response**

Contact the system programmer.

**System programmer response**

Remove the NSS daemon access to the SAF resource, or disable FIPS 140 mode for the NSS daemon. See the NssConfig statement in z/OS Communications Server: IP Configuration Reference for information about the FIPS140 parameter.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: Network Security Server

**Module**

NssConfiguration.cpp

**Routing code**

11

**Descriptor code**

7

**Automation**

This message is output to the MVS console and to syslog.

**Example**

Not applicable.

**EZD1384W**  URL value on statement_type statement for label label was specified on a previous CertificateURL statement
Explanation
The network security services (NSS) server, while processing a CertificateURL or CertificateBundleURL statement in the NSS configuration file, recognized that the URL on that statement matches the URL that was specified on a previous CertificateURL statement. Because CertificateURL statements represent a single certificate, the URL on each of the CertificateURL statements must be unique.

In the message text:

**statement_type**
This value indicates whether the duplicate URL was encountered on a CertificateURL or CertificateBundleURL statement.

**label**
The label on the CertificateURL or CertificateBundleURL statement that contained the duplicate URL value.

System action
Configuration file processing continues and the NSS server continues.

Operator response
Contact the system programmer.

System programmer response
If the retrieval of one of the certificates or certificate bundles is failing because the wrong URL is specified, locate the duplicate URLs in the NSS configuration file and remove or correct the statement that contains the duplicate URL, then use the MODIFY command to refresh the NSS server configuration. See the information about the network security services (NSS) server information in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssStatementParser.cpp

Routing code
10

Descriptor code
12

Automation
This message is written to syslog only.
Example

EZD1384W URL value on CertificateURL statement for label cert3 was specified on a previous CertificateURL statement

EZD1385W URL value on CertificateURL statement for label label_value was specified on a previous CertificateURL or CertificateBundleURL statement

Explanation

While it was processing a CertificateURL statement in the NSS configuration file, the network security services (NSS) server recognized that the URL on the CertificateURL statement matched the URL that was specified on a previous CertificateURL or CertificateBundleURL statement. Because CertificateURL statements represent a single certificate, the URL on each of the CertificateURL statements must be unique.

In the message text:

label_value

The value of the label on the CertificateURL or CertificateBundleURL statement that contained the duplicate URL value.

System action

The processing of the configuration file continues and the NSS server continues.

Operator response

Contact the system programmer.

System programmer response

If the retrieval of one of the certificates or certificate bundles is failing because the wrong URL is specified, locate the duplicate URLs in the NSS configuration file and either remove or correct the statement that contains the duplicate URL and then refresh the NSS server configuration using the MODIFY command. See the information about the network security services (NSS) server information in z/OS Communications Server: IP Configuration Reference for information about the NSSD configuration file.

User response

Not applicable.

Problem determination

None.

Source

z/OS Communications Server TCP/IP: Network Security Server

Module

NssStatementParser.cpp

Routing code

10

Descriptor code

12
Automation
This message is written to syslog only.

Example

EZD1385W URL value on CertificateURL statement for label cert3 was specified on a previous CertificateURL or CertificateBundleURL statement

EZD1386I DISPLAY NSS CONFIGURATION:

Explanation
The Network Security Server (NSS) daemon received the MODIFY NSSproc,DISPLAY subcommand.
See the information about the NSS MODIFY command in z/OS Communications Server: IP System Administrator's Commands.

System action
The Network Security Server daemon (NSS) continues processing the MODIFY DISPLAY command.

Operator response
None.

System programmer response
None.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssModifyCommandHandler.cpp

Routing code
2

Descriptor code
5,8,9

Automation
This message is written to both the operator console and syslog.

Example

EZD1386I DISPLAY NSS CONFIGURATION:
Certificate (label) contains a key that is too short for FIPS 140 mode. Certificate unavailable for the IPSec discipline

Explanation

The Network Security Services (NSS) server is configured to run in a mode that supports Federal Information Processing Standard 140 (FIPS 140). The NSS server detected that a certificate with the specified label contains an RSA key that is not allowed in FIPS 140 mode. The certificate will not be available for the NSS IPSec certificate service. See the information about FIPS 140 and IP security in z/OS Communications Server: IP Configuration Guide.

In the message text:

**label**

The label of the certificate.

System action

NSS server processing continues.

Operator response

None.

System programmer response

If FIPS 140 support is required and the certificate is required for the IPSec discipline, rekey the certificate with an RSA key that has a key size of 1024 bits or greater. If using IKEv2 you can rekey the certificate using an ECDSA key of any length instead of an RSA key. If FIPS 140 support is not required for the NSS server, stop the server, configure FIPS140 No in the NSS server configuration file, and restart the server.

User response

Not applicable.

Problem determination

None.

Source

z/OS Communications Server TCP/IP: Network Security Server

Module

CertRepository.cpp

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.
Example

EZD1387I Certificate (Certificate512) contains a key that is too short for FIPS 140 mode.
Certificate unavailable for the IPSec discipline

EZD1388E  ICSF FIPS mode services are currently unavailable to the NSS daemon

Explanation
The network security services (NSS) server is configured in Federal Information Processing Standards (FIPS) mode, and the server detected that Integrated Cryptographic Services Facility (ICSF) FIPS mode services are unavailable. The NSS server uses ICSF FIPS mode services for various cryptographic functions. When ICSF FIPS mode services are not correctly enabled, the NSS server will not be fully functional to NSS clients.

See the information about network security services in z/OS Communications Server: IP Configuration Guide for information about which NSS server functions require ICSF.

See the information about the network security services (NSS) server information in z/OS Communications Server: IP Configuration Reference for information about the FIPS140 keyword.

System action
The NSS server continues to run, but cannot provide the services that rely on ICSF FIPS mode services.

Operator response
Contact the system programmer.

System programmer response
Configure and start ICSF with FIPS mode enabled. The NSS server automatically detects when ICSF FIPS mode services become available. See the z/OS Cryptographic Services ICSF Administrator's Guide for more information. Alternatively, change the NSS server configuration to disable FIPS mode, and restart the NSS server.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssAnchor.cpp

Routing code
3

Descriptor code
3

Automation
This message goes to the console and to syslog.
EZD1388E ICSF FIPS mode services are currently unavailable to the NSS daemon

**EZD1389I**  DISPLAY NSS URLCACHE:

**Explanation**
The Network Security Server (NSS) daemon received the MODIFY NSSproc,DISPLAY,URLCACHE subcommand. See the information about the **NSS MODIFY command** in z/OS Communications Server: IP System Administrator's Commands.

**System action**
The NSS daemon continues processing the MODIFY DISPLAY,URLCACHE command.

**Operator response**
None.

**System programmer response**
None.

**User response**
No action needed.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssModifyCommandHandler.cpp

**Routing code**
2

**Descriptor code**
5,8,9

**Automation**
This message is written to both the operator console and syslog.

**Example**

EZD1389I  DISPLAY NSS URLCACHE:

EZD1390I  ICSF services are currently unavailable to the NSS daemon operating in FIPS 140 mode
Explanation
The NSS daemon has been configured for FIPS 140 mode and is initializing. ICSF is not currently active. The NSS daemon requires services from ICSF when configured for FIPS 140 mode.

System action
The NSS daemon ends.

Operator response
Contact the system programmer.

System programmer response
The NSS daemon fails to initialize if it is configured for FIPS 140 mode and ICSF is not active.
If you want the NSS daemon to operate in FIPS 140 mode, start ICSF and then restart the NSS daemon.
If you do not want the NSS daemon to operate in FIPS 140 mode, specify "FIPS140 No" in the IPSecDisciplineConfig section of the NSS configuration file and restart the NSS daemon.

User response
None.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp

Routing code
2, 8

Descriptor code
12

Automation
This message is sent to the system console and to syslogd.

Example
EZD1390I ICSF services are currently unavailable to the NSS daemon operating in FIPS 140 mode
EZD1391I FIPS 140 mode is configured for the IPSec Discipline of the NSS daemon

Explanation
"FIPS 140 Yes" has been specified in the configuration file of the NSS daemon.
**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
None.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
NssMainThread.cpp

**Routing code**
*

**Descriptor code**
*

**Automation**
This message is sent to syslogd.

**Example**
EZD1391I FIPS 140 mode is configured for the IPSec Discipline of the NSS daemon

**EZD1392I**  ICSF services are currently available to the NSS daemon

**Explanation**
The NSS daemon is initializing and ICSF is active.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Network Security Server

Module
NssMainThread.cpp

Routing code
2, 8

Descriptor code
12

Automation
This message is sent to the system console and to syslogd.

Example

EZD1392I ICSF services are currently available to the NSS daemon

EZD1526I Connect error for server daemon connection : function errno (description) errnojr errnojr

Explanation
This message is displayed by the ipsec UNIX command when an error occurred during a connect to the specified daemon.

In the message text:

server
The daemon (DM, IKE, or NSS) to which the ipsec command is trying to connect.

function
The name of the C/C++ run-time library function that detected the error.

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errno) information in z/OS UNIX System Services Messages and Codes.

description
Describes the meaning of the errno value.

ernojr
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The ipsec command processing ends.
Operator response
Not applicable.

System programmer response
Ensure that the specified daemon is running and use the function, errno, and errnojr values to fix the problem.

User response
Contact the system programmer.

Problem determination
Determine whether the daemon is running by issuing the MODIFY procname, DISPLAY command where the procname value is the member name of the cataloged procedure used to start the daemon.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
IPsecNMI_Transaction.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Example
This error occurs when the ipsec command tries to connect to a daemon that is not running.

```
EZD1526I Connect error for NSSD connection : connect errno 1128 (EDC8128I Connection refused.)
errnojr 0x120D0253
```

EZD1527I Send error on server daemon connection : function errno errno (description) errnojr errnojr

Explanation
This message is displayed by the ipsec UNIX command when an error occurs during a write operation on the connection to the specified daemon.

In the message text:

**server**
The daemon (DM, IKE, or NSS) to which the ipsec command is trying to write a message.

**function**
The name of the C/C++ run-time library function that detected the error.

**errno**
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

**description**
Describes the meaning of the errno value.
**errnojr**
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
The ipsec command processing ends.

**Operator response**
Not applicable.

**System programmer response**
Ensure that the specified daemon is running and use the `function`, `errno`, and `errnojr` values to fix the problem.

**User response**
Contact the system programmer.

**Problem determination**
Determine whether the daemon is running by issuing the MODIFY `procname`, DISPLAY command where the `procname` value is the member name of the cataloged procedure used to start the daemon.

**Source**
z/OS Communications Server: z/OS UNIX ipsec command

**Module**
IPsecNMI_Transaction.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Example**
This error can occur when the NSS or IKE daemon is stopped while an ipsec command request is in progress.

```
EZD1527I Send error on NSSD connection : write errno 1124 (EDC8124I Socket not connected.)
errnojr 0x120D025C
```

**EZD1528I Receive error on server daemon connection : function errnoerno (description) errnojr errnojr**

**Explanation**
This message is displayed by the ipsec UNIX command when an error occurs during a read operation on the connection from the server daemon.

In the message text:

**server**
The daemon (DM, IKE, or NSS) from which the ipsec command is trying to read a message.

**function**
The name of the C/C++ run-time library function that detected the error.
**errno**
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

**description**
Describes the meaning of the errno value.

**errnojr**
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
The ipsec command processing ends.

**Operator response**
Not applicable.

**System programmer response**
Ensure that the specified daemon is running and use the function, errno, and errnojr values to fix the problem.

**User response**
Contact the system programmer.

**Problem determination**
Determine whether the daemon is running by issuing the MODIFY procname, DISPLAY command where the procname value is the member name of the cataloged procedure used to start the daemon.

**Source**
z/OS Communications Server: z/OS UNIX ipsec command

**Module**
IPsecNMI_Transaction.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Example**
This error can occur when the NSS or IKE daemon is stopped while an ipsec command request is in progress.

```
EZD1528I Receive error on NSSD connection : read errno 1124 (EDC8124I Socket not connected.)
errnojr 0x120D025C
```

**EZD1529I**  **Data received over the server daemon connection is not valid : return code returncode**

**Explanation**
This message is displayed by the ipsec UNIX command when data received from the specified daemon is not in the expected format.
In the message text:

**server**
The daemon with the connection that received the illegal data. The possible values are DM, IKE, or NSS.

**returncode**
Possible values are:

1. The message header identifier is not valid.
2. The message header version number is not valid.
3. The message type is not valid.
4. The message header size is not valid.
5. The message size is not valid.
6. The message contains a reserved area that is not set to zeros.
7. The message contains a record length that is not valid.
8. The message contains a record count that is not valid.
9. The message contains a section with a length that is not valid.
10. The message contains a section with a count field that is not valid.
11. The response message contains a correlation ID that does not match the correlation ID in the request message.
12. The response message contains a message type that does not correspond to the message type in the request message.

**System action**
The ipsec command processing ends.

**Operator response**
Not applicable.

**System programmer response**
Run the ipsec command again with the -d option to determine the sequence of events leading up to the error. Contact the IBM Software Support Center.

**User response**
If the error persists, contact the system programmer.

**Problem determination**
Not applicable.
Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
IPsecNMI_Transaction.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Example
The response message received from the daemon does not correspond to the request message sent to the daemon.

EZD1529I Data received over the NSSD connection is not valid: return code 14

EZD1530I  Message received from server daemon with return code returncode (description) reason code reasoncode

Explanation
The server daemon returned a response message with a nonzero return code.

In the message text:
server
The daemon (DM, IKE, or NSS) from which the ipsec command received the message.

returncode
The return code contained in the response message. These return codes are listed and described in the network manager return and reason codes information in z/OS Communications Server: IP Programmer's Guide and Reference and in the diagnosing network security services (NSS) server problems information in z/OS Communications Server: IP Diagnosis Guide.

description
Describes the meaning of the return code.

reasoncode
The reason code contained in the response message. These reason codes are listed and described in the network manager return and reason codes information in z/OS Communications Server: IP Programmer's Guide and Reference and in the diagnosing network security services (NSS) server problems information in z/OS Communications Server: IP Diagnosis Guide. For reason codes with a mnemonic starting with NSS look first in the z/OS Communications Server: IP Diagnosis Guide. For reason codes with a mnemonic starting with the letters NMs look first in the z/OS Communications Server: IP Programmer's Guide and Reference.

System action
The ipsec command processing ends.

Operator response
Not applicable.
**System programmer response**

See the information about the network manager return and reason codes in z/OS Communications Server: IP Programmer's Guide and Reference and the diagnosing network security services (NSS) server problems in z/OS Communications Server: IP Diagnosis Guide to determine the appropriate response.

**User response**

Contact the system programmer.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server: z/OS UNIX ipsec command

**Module**

IPsecNMI_Transaction.cpp

**Routing code**

Not applicable.

**Descriptor code**

Not applicable.

**Example**

```
EZD1530I Message received from NSSD with return code 121 ( EDC5121I  Invalid argument. )
reason code NSSRsnUnknownClientName
EZD1531I  Internal ipsec command error : return code returncode
```

**EZD1531I Internal ipsec command error : return code returncode**

**Explanation**

This message is displayed by the ipsec UNIX command when an internal error is detected.

In the message text:

- **returncode**
  - The return code from the ipsec command.

**System action**

The ipsec command processing ends

**Operator response**

Not applicable.

**System programmer response**

Run the ipsec command again with the -d option to determine the sequence of events leading up to the error. Contact the IBM Software Support Center.

**User response**

Contact the system programmer.
Problem determination
See the system programmer response.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
IPsecNMI_Record.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Example

| EZD1531I Internal ipsec command error : return code 1 |

EZD1532I  Client *clientname* is not available

Explanation
The network security services (NSS) client specified with the ipsec -z option cannot be found.

In the message text:

*clientname*
   The name of an NSS client specified with the -z option of the ipsec UNIX command.

System action
The ipsec command processing ends.

Operator response
Not applicable.

System programmer response
Determine whether there is an NssStackConfig statement defining the client in the IKE daemon configuration file on the system where the client is running. See the information about IP security in z/OS Communications Server: IP Configuration Guide and the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference for information about configuring the NssStackConfig statement.

User response
Determine whether the NSS client is connected to the NSS server by issuing the ipsec -x display UNIX command on the system where NSS is running. Determine whether the *clientname* value is the name of a valid NSS client by issuing the ipsec -w display UNIX command on the system where the client is running. If the name is not valid, contact the system programmer.

Problem determination
None.
Example

EZD1532I Client MVS134_TCPCS3 is not available

EZD1533I Only one defensive filter name may follow the -N option when used together with the -F function function

Explanation
More than one defensive filter name was specified for the -F function identified in the message. Only one defensive filter name is allowed.

In the message text:

function
The -F function specified by the user on the ipsec command.

System action
No action is taken for any of the defensive filters specified in the -N option. The ipsec command processing ends.

Operator response
Reissue the ipsec command, specifying only one defensive filter name. If the action needs to be applied to multiple defensive filters, issue multiple ipsec commands.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the man ipsec command in a z/OS UNIX shell to obtain information about the ipsec command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command
Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1533I Only one defensive filter name may follow the -N option when used together with the -F Add function

EZD1534I  The keyword keyword was specified multiple times

Explanation
The keyword identified in the message was specified more than once on the `ipsec` command. The keyword is allowed only once.

In the message text:

`keyword`
   The keyword that was specified multiple times on the `ipsec` command.

System action
The `ipsec` command processing ends.

Operator response
Reissue the `ipsec` command, specifying the keyword once.

See the information about managing network security in `z/OS Communications Server: IP System Administrator's Commands` or issue the `man ipsec` command in a `z/OS UNIX` shell to obtain information about the `ipsec` command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
`z/OS Communications Server: z/OS UNIX ipsec command`
Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>EZD1534I The srcip keyword was specified multiple times</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1535I No value specified with the keyword keyword</td>
</tr>
</tbody>
</table>

Explanation
A keyword is specified without a value on the `ipsec` command.

In the message text:

`keyword`
The keyword specified on the `ipsec` command that is missing a value.

System action
The `ipsec` command processing ends.

Operator response
Reissue the `ipsec` command, specifying a value for the keyword identified in the message.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
Not applicable.
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1535I No value specified with the srcip keyword

EZD1536I ALL is not a valid value for the -N option in this context

Explanation
ALL was specified as the value for the -N option on the \texttt{ipsec} command in a context that is not valid. The -N ALL option is allowed only when deleting all defensive filters with ipsec -F delete.

System action
The \texttt{ipsec} command processing ends

Operator response
Reissue the \texttt{ipsec} command, specifying a valid value for the -N option.

See the information about managing network security in \textit{z/OS Communications Server: IP System Administrator's Commands} or issue the \texttt{man ipsec} command in a z/OS UNIX shell to obtain information about the \texttt{ipsec} command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
\textit{z/OS Communications Server: z/OS UNIX ipsec command}

Module
Not applicable.

Routing code
10

Descriptor code
12
Automation
Not applicable.

Example

EZD1536I ALL is not a valid value for the -N option in this context

EZD1537I  value is not a valid keywordname value

Explanation
An incorrect value is specified for a keyword on the ipsec command.

In the message text:

value  The incorrect value specified on the ipsec command.

keywordname  The keyword that has the incorrect value.

System action
The ipsec command processing ends

Operator response
Reissue the ipsec command, specifying a valid value for the keyword identified in the message.

See the information about managing network security in z/OS Communications Server: IP System Administrator’s Commands or issue the man ipsec command in a z/OS UNIX shell to obtain information about the ipsec command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
Not applicable.

Routing code
10

Descriptor code
12
Automation
Not applicable.

Example
EZD1537I 500000 is not a valid srcport value

EZD1538I Option keyword1 value1 conflicts with option keyword2 value2

Explanation
There is a conflict between the values specified for keyword1 and keyword2 on the ipsec command.

In the message text:

keyword1
A keyword specified on the ipsec command.

value1
The value specified for keyword1 on the ipsec command.

keyword2
A second keyword specified on the ipsec command.

value2
The value specified for keyword2 on the ipsec command.

System action
The ipsec command processing ends

Operator response
Reissue the ipsec command, correcting the conflict identified in the message.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the man ipsec command in a z/OS UNIX shell to obtain information about the ipsec command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
Not applicable.

Routing code
10
**Example**

```
EZD1538I Option prot opaque conflicts with option routing either
```

**Explanation**
The protocol value is not valid with the address type (IPv4 or IPv6) on the `ipsec` command.

In the message text:

- **protvalue**
  - The protocol value specified on the `ipsec` command.

- **addresstype**
  - The IP address type of the defensive filter. Possible values are IPv4 or IPv6. The IP address type is determined by the `srcip` and `destip` keywords.

**System action**
The `ipsec` command processing ends.

**Operator response**
Reissue the `ipsec` command, correcting the conflict identified by the message.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server: z/OS UNIX ipsec command

**Module**
Not applicable.

---

**Routing code**
10
Example

EZD1539I opaque conflicts with IPv4

EZD1540I Defensive filter filtername successfully added to stack stackname

Explanation

This message is received in response to an `ipsec -F add` command that is directed to a specific TCP/IP stack. This message confirms that the defensive filter was added.

In the message text:

*filtername*
   The name of the filter that was added.

*stackname*
   The name of the TCP/IP stack to which the defensive filter was added.

System action

The z/OS UNIX `ipsec` command processing completes successfully.

Operator response

None.

System programmer response

None.

User response

Not applicable.

Problem determination

None.

Source

z/OS Communications Server: z/OS UNIX ipsec command

Module

Not applicable.

Routing code

10

Descriptor code

12
EZD1541I  Global defensive filter *filtername* successfully added

**Explanation**

This message is received in response to an `ipsec -G -F add` command. This message confirms that the global defensive filter was successfully added to all eligible TCP/IP stacks on the z/OS image. An eligible stack is a stack that is enabled for IP security and defined with a mode of active or simulate in the Defense Manager daemon (DMD) configuration file.

In the message text:

*filtername*

The name of the filter that was added.

See the information about global and stack-specific defensive filters in z/OS Communications Server: IP Configuration Guide for information about global defensive filter add processing.

**System action**

The z/OS UNIX `ipsec` command processing completes successfully.

**Operator response**

None.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server: z/OS UNIX ipsec command

**Module**

Not applicable.

**Routing code**

10

**Descriptor code**

12
**Automation**
Not applicable.

**Example**

| EZD1541I Global defensive filter Global_Block_inbound_10.1.1.1 successfully added |

**EZD1542I** All defensive filters successfully deleted from stack `stackname`

**Explanation**

This message is received in response to an `ipsec -F delete -N all` command that is directed to a specific TCP/IP stack. This message confirms that all defensive filters were deleted from the stack indicated by `stackname`.

In the message text:

*stackname*

The name of the TCP/IP stack from which the defensive filters were deleted.

See the information about global and stack-specific defensive filters in z/OS Communications Server: IP Configuration Guide for information about defensive filter delete processing.

**System action**
The z/OS UNIX `ipsec` command processing completes successfully.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server: z/OS UNIX ipsec command

**Routing code**
10

**Descriptor code**
12
EZD1543I All global defensive filters successfully deleted

Explanation
This message is received in response to an `ipsec -F delete -N all -G` command. This message confirms that all global defensive filters were deleted.

See the information about global and stack-specific defensive filters in z/OS Communications Server: IP Configuration Guide for information about global defensive filter delete processing.

System action
The z/OS UNIX `ipsec` command processing completes successfully.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command
EZD1544I Defensive filter filtername was successfully deleted from stack stackname

Explanation
This message is received in response to an `ipsec -F delete` command that is directed to a specific TCP/IP stack. This message confirms that the defensive filter was deleted from the stack indicated by `stackname`.

In the message text:

**filtername**
- The name of the filter that was deleted.

**stackname**
- The name of the TCP/IP stack from which the defensive filter was deleted.

See the information about global and stack-specific defensive filters in z/OS Communications Server: IP Configuration Guide for information about defensive filter delete processing.

System action
The z/OS UNIX `ipsec` command processing completes successfully.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

```
EZD1544I Defensive filter Block_inbound_10.1.1.1 was successfully deleted from stack TCPCS
```

EZD1545I Global defensive filter filtername was successfully deleted
Explanation
This message is received in response to an `ipsec -F delete -G` command. This message confirms that the global defensive filter was successfully deleted.

In the message text:

*filename*
The name of the filter that was deleted.

See the information about global and stack-specific defensive filters in z/OS Communications Server: IP Configuration Guide for information about global defensive filter delete processing.

System action
The z/OS UNIX `ipsec` command processing completes successfully.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

```
EZD1545I Global defensive filter Block_inbound_10.1.1.1_Global was successfully deleted
```

```
EZD1546I Defensive filter *filename* was not found in stack *stackname*
```

1006  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Explanation
This message is received in response to an `ipsec -F delete` command directed to a specific TCP/IP stack. This message indicates that the defensive filter was not found in the stack.

In the message text:

**filtername**
The name of the filter specified by the `-N` option on the `ipsec` command.

**stackname**
The name of the TCP/IP stack to which the `ipsec` command was directed.

System action
The z/OS UNIX `ipsec` command processing ends.

Operator response
If the filter name specified on the `ipsec` command was incorrect, reissue the `ipsec` command, specifying the correct filter name. Issue the `ipsec -F display` command to display the defensive filters installed in the stack.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

```
EZD1546I Defensive filter filter1 was not found in stack TCPCS
```
Global defensive filter *filtername* was not found

Explanation

This message is received in response to an `ipsec -F delete -G` command. This message indicates that the global defensive filter was not found.

In the message text:

*filtername*

The name of the filter specified on the `ipsec` command by the `-N` option.

System action

The z/OS UNIX `ipsec` command processing ends.

Operator response

If the filter name specified on the `ipsec` command was incorrect, reissue the `ipsec` command, specifying the correct filter name. Issue the `ipsec -F display -G` command to display the global defensive filters. If the filter that you want to delete is not a global filter, reissue the `ipsec` command without the `-G` option.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the `ipsec` command syntax and options.

System programmer response

None.

User response

Not applicable.

Problem determination

None.

Source

z/OS Communications Server: z/OS UNIX ipsec command

Module

Not applicable.

Routing code

10

Descriptor code

12

Automation

Not applicable.

Example

```
EZD1547I Global defensive filter filter1_global was not found
```

1008 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
EZD1548I Defensive filter *filtername* was successfully updated in stack *stackname*

**Explanation**

This message is received in response to an **ipsec -F update** command that is directed to a specific TCP/IP stack. This message confirms that the defensive filter was updated in the stack indicated by *stackname*.

In the message text:

*filtername*

The name of the filter that was updated.

*stackname*

The name of the TCP/IP stack in which the defensive filter was updated.

See the information about global and stack-specific defensive filters in *z/OS Communications Server: IP Configuration Guide* for information about defensive filter update processing.

**System action**

The z/OS UNIX **ipsec** command processing completes successfully.

**Operator response**

None.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server: z/OS UNIX ipsec command

**Module**

Not applicable.

**Routing code**

10

**Descriptor code**

12

**Automation**

Not applicable.

**Example**

```
EZD1548I Defensive filter Block_inbound_10.1.1.1 was successfully updated in stack TCPCS
```

EZD1549I Global defensive filter *filtername* was successfully updated
Explanation
This message is received in response to an `ipsec -G -F update` command. This message confirms that the global defensive filter was successfully updated.

In the message text:

*filename*
   The name of the filter that was updated.

See the information about global and stack-specific defensive filters in z/OS Communications Server: IP Configuration Guide for information about global defensive filter update processing.

System action
The z/OS UNIX `ipsec` command processing completes successfully.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

`EZD1549I Global defensive filter Block_inbound_10.1.1.1_Global was successfully updated`

`EZD1550I Defense Manager daemon reported an error - error_description`
Explanation

This message is received in response to an `ipsec -F` command. The Defense Manager daemon (DMD) detected an error and reported it to the `ipsec` command processor.

In the message text:

**error_description**

A description of the error that the DMD reported. Possible values are:

<table>
<thead>
<tr>
<th><code>error_description</code> value</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>duplicate filter name</td>
<td>This error description is received in response to an <code>ipsec -F add</code> command that specifies a filter name with the -N option that conflicts with an unexpired defensive filter name.</td>
</tr>
<tr>
<td>filter is not found</td>
<td>This error description is received in response to an <code>ipsec -F update</code> command when the filter to be updated is not found.</td>
</tr>
<tr>
<td>stack is not configured for IPSECURITY</td>
<td>This error description is received in response to an <code>ipsec -F add</code> command directed to a specific stack when the stack is not configured for IP Security.</td>
</tr>
<tr>
<td>stack is not configured for IPv6 IPSECURITY</td>
<td>This error description is received in response to an <code>ipsec -F add</code> command for an IPv6 filter that is directed to a specific stack and the stack is not configured for IPv6 Security.</td>
</tr>
<tr>
<td>stack is not configured</td>
<td>This error description is received in response to an <code>ipsec -F add</code> command directed to a specific stack when the stack is not configured with a DmStackConfig statement in the DMD configuration file.</td>
</tr>
<tr>
<td>stack mode is INACTIVE</td>
<td>This error description is received in response to an <code>ipsec -F add</code> command directed to a specific stack when the stack is configured as Inactive on the DmStackConfig statement in the DMD configuration file.</td>
</tr>
<tr>
<td>user ID is not authorized</td>
<td>This error description is received in response to an <code>ipsec -F</code> command when the user ID is not authorized to the EZB.IPSECCMD profiles through the security access facility.</td>
</tr>
<tr>
<td>too many connections</td>
<td>This error description is received in response to an <code>ipsec -F</code> command when the <code>ipsec</code> command is unable to process the request because it is has reached its limit of concurrent <code>ipsec</code> command connections.</td>
</tr>
<tr>
<td>log no is not allowed for a filter with mode simulate</td>
<td>This error description is received in response to an <code>ipsec -F update log no</code> command for an existing defensive filter with a mode of simulate. Logging cannot be turned off for a filter with a mode of simulate.</td>
</tr>
</tbody>
</table>
**error_description value** | **Explanation**
--- | ---
bad data | These error descriptions are received in response to an **ipsec -F** command when an internal error has occurred.
IOCTL failure | connection ID is not valid
internal error is reported by the Defense Manager daemon | memory allocation error
cannot retrieve user ID credentials | client is already connected

**System action**

The z/OS UNIX **ipsec** command processing ends.

**Operator response**

The operator response is based on the **error_description** value as shown in the following table.

<table>
<thead>
<tr>
<th><strong>error_description value</strong></th>
<th><strong>Operator response</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>duplicate filter name</td>
<td>Reissue the <strong>ipsec -F add</strong> command with a unique defensive filter name. See the information about defensive filter names in z/OS Communications Server: IP Configuration Guide for more information.</td>
</tr>
<tr>
<td>filter is not found</td>
<td>If the filter name specified on the <strong>ipsec</strong> command was incorrect, reissue the <strong>ipsec</strong> command, specifying the correct filter name. Issue the <strong>ipsec -F display</strong> command to display the defensive filters installed in the stack. See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the <strong>man ipsec</strong> command in a z/OS UNIX shell to obtain information about the <strong>ipsec</strong> command syntax and options.</td>
</tr>
<tr>
<td>stack is not configured for IPSECURITY</td>
<td>Contact the system programmer.</td>
</tr>
<tr>
<td>stack is not configured for IPv6 IPSECURITY</td>
<td>Contact the system programmer.</td>
</tr>
<tr>
<td>stack is not configured</td>
<td>Contact the system programmer.</td>
</tr>
<tr>
<td>stack mode is INACTIVE</td>
<td>Contact the system programmer.</td>
</tr>
<tr>
<td>user ID is not authorized</td>
<td>Contact the system programmer.</td>
</tr>
<tr>
<td>too many connections</td>
<td>Reissue the <strong>ipsec</strong> command. If the command continues to fail, contact the system programmer.</td>
</tr>
<tr>
<td>log no is not allowed for a filter with mode simulate</td>
<td>None.</td>
</tr>
<tr>
<td>error_description value</td>
<td>Operator response</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>bad data</td>
<td>Reissue the <code>ipsec</code> command. If the command continues to fail, contact the system programmer.</td>
</tr>
<tr>
<td>IOCTL failure</td>
<td></td>
</tr>
<tr>
<td>connection ID is not valid</td>
<td></td>
</tr>
<tr>
<td>internal error is reported by the Defense Manager daemon</td>
<td></td>
</tr>
<tr>
<td>memory allocation error</td>
<td></td>
</tr>
<tr>
<td>cannot retrieve user ID credentials</td>
<td></td>
</tr>
<tr>
<td>client is already connected</td>
<td></td>
</tr>
</tbody>
</table>

**System programmer response**

The system programmer response is based on the `error_description` value as shown in the following table.

<table>
<thead>
<tr>
<th>error_description value</th>
<th>System programmer response</th>
</tr>
</thead>
<tbody>
<tr>
<td>duplicate filter name</td>
<td>None.</td>
</tr>
<tr>
<td>filter is not found</td>
<td>None.</td>
</tr>
</tbody>
</table>
| stack is not configured for IPSECURITY           | If you want to allow defensive filters to be installed for the TCP/IP stack, enable IP security.  
|                                                 | See the information about enabling the IP security function in z/OS Communications Server: IP Configuration Guide. |
| stack is not configured for IPv6 IPSECURITY      | If you want to allow IPv6 defensive filters to be installed for the TCP/IP stack, enable IP security for IPv6.  
|                                                 | See the information about enabling the IP security function in z/OS Communications Server: IP Configuration Guide. |
| stack is not configured                          | If you want to allow defensive filters to be installed for the TCP/IP stack, configure a DmStackConfig statement for the stack in the DMD configuration file.  
|                                                 | The Mode keyword must be set to Active or Simulate to enable defensive filtering.  
<p>|                                                 | See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. |</p>
<table>
<thead>
<tr>
<th>error_description value</th>
<th>System programmer response</th>
</tr>
</thead>
</table>
| stack mode is INACTIVE  | If you want to allow defensive filters to be installed for the TCP/IP stack, the defensive filtering mode for the stack must be Active or Simulate. If the DmStackConfig statement for this stack in the DMD configuration file has Mode Inactive specified, update the mode to Active or Simulate. Issue the MODIFY REFRESH command to begin using the new value.  
If the mode is already Active or Simulate, a MODIFY FORCE_INACTIVE command might have been issued, forcing defensive filtering to Inactive. Issue a MODIFY procname,REFRESH,FILE=file command to enable defensive filtering.  
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. |
| user ID is not authorized| Create the required SERVAUTH profiles to authorize the user ID that is issuing the `ipsec` command. If the SERVAUTH profiles exist, give the user ID that is issuing the `ipsec` command permission to access the profiles.  
See the information about ipsec command security in z/OS Communications Server: IP System Administrator's Commands for more information about the required SERVAUTH profiles. |
| too many connections    | This error might be received as the result of automation that is attempting to add, update, or delete a large number of defensive filters simultaneously. Update the automation to issue the `ipsec` commands sequentially. If automation is not being used, this error might be the result of an internal error. For an internal error, contact IBM software support services. Provide a dump of the DMD. If available, provide CTRACE information for component SYSTCPDM. |
| log no is not allowed for a filter with mode simulate | None. |
| bad data                | Contact IBM software support services. Provide a dump of the DMD. If available, provide CTRACE information for component SYSTCPDM. |
| IOCTL failure           | |
| connection ID is not valid | |
| internal error is reported by the Defense Manager daemon | |
| memory allocation error | |
| cannot retrieve user ID credentials | |
| client is already connected | |
User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server z/OS UNIX **ipsec** command

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

```
EZD1550I Defense Manager daemon reported an error - duplicate filter name
```

**EZD1551I**  
*option1* is valid only when used with the *option2* primary option

Explanation
An option on the **ipsec** command is valid only with a specific primary option. The **ipsec** command has been issued with a combination of options that is not valid or the primary option is missing.

In the message text:

**option1**
- An option specified on the **ipsec** command.

**option2**
- The primary option on the **ipsec** command for which *option1* is valid.

System action
The z/OS UNIX **ipsec** command processing ends.

Operator response
Reissue the **ipsec** command with a valid combination of options.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the `man ipsec` command in a z/OS UNIX shell to obtain information about the **ipsec** command syntax and options.

System programmer response
None.
User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
<table>
<thead>
<tr>
<th>EZD1551I -G is valid only when used with the -F primary option</th>
</tr>
</thead>
</table>

EZD1552I The keyword keyword is not valid in the context in which it appears

Explanation
A valid keyword is specified out of context on the ipsec -F add or ipsec -F update command.

In the message text:

keyword
The keyword specified out of context on the ipsec command.

System action
z/OS UNIX ipsec command processing ends.

Operator response
Reissue the ipsec command, correcting the error identified in the message.

See the information about managing network security in z/OS Communications Server: IP System Administrator's Commands or issue the man ipsec command in a z/OS UNIX shell to obtain information about the ipsec command syntax and options.

System programmer response
None.

User response
Not applicable.
Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX ipsec command

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
In the following example, the fragmentsonly keyword is specified without specifying routing routed.

ipsec -F add srcip 10.1.1.1 fragmentsonly yes -N filter1

This example command is incorrect and results in the following message:

EZD1552I The keyword fragmentsonly is not valid in the context in which it appears

EZD1553I  keyword is not a valid keyword

Explanation
An incorrect keyword was specified on the ipsec -F add or ipsec -F update command.

In the message text:

keyword
The incorrect keyword specified on the ipsec command.

System action
The ipsec command processing ends

Operator response
Reissue the ipsec command, specifying a valid keyword for the incorrect keyword identified in the message.

See the information about managing network security in z/OS Communications Server: IP System Administrator’s Commands or issue the man ipsec command in a z/OS UNIX shell to obtain information about the ipsec command syntax and options.

System programmer response
None.
**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server: z/OS UNIX ipsec command

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

```
EZD1553I protocol is not a valid keyword
```

**EZD1576  PAGENT IS READY FOR SERVICES CONNECTION REQUESTS**

**Explanation**
This message is issued when the ServicesConnection configuration statement is specified in the Policy Agent main configuration file and the Policy Agent is ready to provide connections for its services requestors.

See the information about the ServicesConnection statement in z/OS Communications Server: IP Configuration Reference for information about configuring the ServicesConnection statement.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.
**Source**
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

**Module**
plfmmisc

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to the system console and syslog on the Policy Agent. You might want to automate on this message to determine when the Policy Agent is ready to provide services for import requestors.

**Example**
Not applicable.

**EZD1577**  PAGENT SERVICESCONNECTION STATEMENT CONTAINS ERRORS

**Explanation**
A ServicesConnection statement in the main configuration file contains errors on the Policy Agent.

**System action**
The Policy Agent continues but does not listen for services requestors using the ServicesConnection statement.

**Operator response**
Contact the system programmer. If the system programmer indicates that more information is required in the Policy Agent log file, restart the Policy Agent with a minimum of LogLevel 127 configured in the main configuration file, or with the `-d 1` start option.

**System programmer response**
Examine the log file to determine the cause of the problem. Correct the Policy Agent configuration errors identified in the log and restart the Policy Agent. If you need more information to diagnose the errors, restart the Policy Agent with a minimum of LogLevel 127 or start the policy server with the `-d 1` start option to see the configuration errors.

See the information about the ServicesConnection statement in z/OS Communications Server: IP Configuration Reference for information about configuring the ServicesConnection statement.

**User response**
Not applicable.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)
Module
plfmnmisc

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and syslog on the Policy Agent. You might want to automate on this message to determine whether there are configuration errors that will prevent import requestors from connecting to the Policy Agent.

Example
Not applicable.

EZD1578 PAGENT IS UNABLE TO PROCESS REQUESTS FROM SERVICES REQUESTORS

Explanation
The Policy Agent is unable to process requests from services requestors. One possible reason is that the information configured on the ServicesConnection configuration statement is incorrect.

See the information about the ServicesConnection statement in z/OS Communications Server: IP Configuration Reference for information about configuring the ServicesConnection statement.

See the information about import requestor connection problems in z/OS Communications Server: IP Diagnosis Guide for possible reasons this message was issued.

System action
The Policy Agent continues but does not respond to services connection requests.

Operator response
Contact the system programmer. If the system programmer indicates that more information is required in the Policy Agent log file, restart the Policy Agent with a minimum of LogLevel 127 configured in the main configuration file, and with the -d 128 start option.

System programmer response
Examine the log file to determine the cause of the problem. If the problem is the result of a socket or bind failure, the problem might be an incorrect port specified on the ServicesConnection configuration statement. Verify that the port is valid and correct the statement if necessary. Otherwise, restart the Policy Agent with the LogLevel 127 and the -d 128 start option and recreate the problem to diagnose the errors.

See the information about gathering diagnostic information about Policy Agent problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

User response
Not applicable.
Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

Module
plfmmisc

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and syslog on the Policy Agent. You might want to automate on this message to determine whether there are problems that will prevent import requestors from connecting to the Policy Agent or from retrieving import policies. You might also want to automate on the following related message:

EZD1576I PAGENT IS READY FOR SERVICES CONNECTION REQUESTS

Example
Not applicable.

EZD1579I PAGENT POLICIES ARE NOT ENABLED FOR image : type

Explanation
The policies indicated by the type value that is defined in a configuration file are not enabled for the TCP/IP stack indicated by the image value. The policies are not enabled because the underlying stack function (for example, AT-TLS or IPSec) is not enabled on the stack.

In the message text:

image
The name of the TCP/IP stack.

type
The policy type that is not enabled. Possible values are:

IPSEC
IP Filtering, KeyExchange and LocalDynVpn policies

TTLS
Application Transparent Transport Layer Security (AT-TLS) policies

System action
The results depend on the type value as follows:

IPSEC
Policy Agent does not parse or read the IPSec configuration file for this image.

TTLS
Policy Agent reads and parses the AT-TLS configuration file and installs the policies, but the TCP/IP stack does not enforce the policies.
If the ServicesConnection statement is configured with Security Secure, then Policy Agent automatically creates an AT-TLS policy for the connection, but the TCP/IP stack does not enforce the policy and the generated AT-TLS policy is not in effect.

**Operator response**

See the system programmer response.

**System programmer response**

The action depends on the type value as follows:

**IPSEC**

If you want IP security to be enabled for IPv4, use the IPCONFIG IPSECURITY statement in the TCP/IP profile to configure the stack for IPv4 IP security. If you want IP security enabled for IPv6, use the IPCONFIG6 IPSECURITY statement in the TCP/IP profile to configure the stack for IPv6 IP security. See the information about IPCONFIG IPSECURITY and IPCONFIG6 IPSECURITY in z/OS Communications Server: IP Configuration Reference and the information about default IP filter policy and IP security Policy in z/OS Communications Server: IP Configuration Guide for more information.

**TTLS**

If you want AT-TLS enabled, configure the stack for AT-TLS using the TTLS parameter on the TCPCONFIG statement in the TCP/IP profile. See the information about the TCPCONFIG statement in z/OS Communications Server: IP Configuration Reference for more information. See the information about the AT-TLS configuration in PROFILE.TCPIP in z/OS Communications Server: IP Configuration Guide for information about how to enable AT-TLS.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

**Module**

plfmmisc.c

**Routing code**

10

**Descriptor code**

12

**Example**

EZD1579I  PAGENT POLICIES ARE NOT ENABLED FOR TCPCS1 :  IPSEC

EZD1580I  PAGENT IS RESTARTING application

**Explanation**

Policy Agent detected that an application that is configured for monitoring on the AutoMonitorApps configuration statement is not running. Policy Agent is preparing to restart the application.
In the message text:

**application**
The name of the application that is to be restarted. If the application runs one instance per stack, the corresponding TCP/IP stack name precedes the application name, separated by a forward slash (/).

**System action**
Policy Agent processing continues.

**Operator response**
Not Applicable.

**System programmer response**
Not Applicable.

**User response**
Not applicable.

**Problem determination**
Not Applicable.

**Source**
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

**Module**
pmonapps

**Routing code**
10

**Descriptor code**
12

**Automation**
Not Applicable.

**Example**

```
EZD1580I PAGENT IS RESTARTING TCPIP1/TRMD
```

**EZD1581I PAGENT IS UNABLE TO START application**

**Explanation**
Policy Agent attempted to start or restart an application that is configured for monitoring on the AutoMonitorApps configuration statement. The application failed to start successfully within the retry limits configured on the AutoMonitorApps configuration statement.

In the message text:
**application**

The name of the application that failed to start. If the application runs one instance per stack, the corresponding TCP/IP stack name precedes the application name, separated by a forward slash (/).

**System action**

Policy Agent processing continues.

**Operator response**

Contact the system programmer.

**System programmer response**

Look for messages issued by the application to determine the reason why the application failed to start successfully. See the information about running the application in *z/OS Communications Server: IP Configuration Reference* to determine whether additional logging or tracing is needed. See the information about the application in *z/OS Communications Server: IP Diagnosis Guide* for additional diagnosis information. When the problem with the application has been resolved, issue a MODIFY `procname`,MON,START,`applname` command to the Policy Agent to start the application and to resume Policy Agent monitoring of the application. See the information about the Policy Agent MODIFY command in *z/OS Communications Server: IP System Administrator's Commands* for details.

**User response**

Not applicable.

**Problem determination**

See the system programmer response.

**Source**

z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

**Module**

pmonapps

**Routing code**

10

**Descriptor code**

12

**Automation**

Not applicable.

**Example**

```
EZD1581I  PAGENT IS UNABLE TO START NSSD

EZD1582I  PAGENT MODIFY COMMAND UNSUCCESSFUL - application ALREADY status
```
Explanation
A MODIFY command was issued to the Policy Agent to start or stop one application or all applications that are configured for monitoring on the AutoMonitorApps configuration statement, but the application or applications were already started or stopped.

In the message text:

*application*
   - The name of the application that was specified on the MODIFY command. If the application runs one instance per stack, the corresponding TCP/IP stack name precedes the application name, separated by a forward slash (/). If the MODIFY command was issued for all applications, the application value is APPLICATIONS.

*status*
   - The status of the application.

*System action*
The MODIFY command is ignored.

*Operator response*
Not applicable.

*System programmer response*
Not applicable.

*User response*
Not applicable.

*Problem determination*
Not applicable.

*Source*
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

*Module*
pzosinit

*Routing code*
10

*Descriptor code*
12

*Automation*
Not applicable.

*Example*
F PAGENT,MON,START,DMD

```
EZD1582I PAGENT MODIFY COMMAND UNSUCCESSFUL - DMD ALREADY STARTED
```
EZD1583I  PAGENT MODIFY COMMAND UNSUCCESSFUL - application NOT CONFIGURED FOR MONITORING

Explanation
A MODIFY command was issued to the Policy Agent to start, restart, or stop an application, but the application is not configured for monitoring on the AutoMonitorApps configuration statement.

In the message text:

application
The name of the application that was specified on the MODIFY command. If the application runs one instance per stack, the corresponding TCP/IP stack name precedes the application name, separated by a forward slash (/).

System action
The MODIFY command is ignored.

Operator response
See the system programmer response.

System programmer response
If you want to monitor the application, configure the application for monitoring. See the information about configuring Policy Agent to automatically monitor applications in z/OS Communications Server: IP Configuration Guide and the AutoMonitorApps statement in z/OS Communications Server: IP Configuration Reference for.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

Module
pzosinit

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
F PAGENT,MON,START,DMD
EZD1583I PAGENT MODIFY COMMAND UNSUCCESSFUL - DMD NOT CONFIGURED FOR MONITORING
EZD1584I  PAGENT IS ALREADY STARTING application

Explanation
A MODIFY command was issued to the Policy Agent to start, restart, or stop an application, but the application is in the process of being started by the Policy Agent.

In the message text:

application
The name of the application that was specified on the MODIFY command. If the application runs one instance per stack, the corresponding TCP/IP stack name precedes the application name, separated by a forward slash (/).

System action
The MODIFY command is ignored.

Operator response
If you issued a MODIFY procname,MON,STOP,app command, then wait for the application to successfully start and then reissue the command.

System programmer response
Not applicable.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

Module
pzosinit

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
F PAGENT,MON,RESTART,TRMD,P=TCPIP3

EZD1584I  PAGENT IS ALREADY STARTING TCPIP3/TRMD

EZD1585I  PAGENT statement STATEMENT CONTAINS ERRORS
Explanation
A configuration statement in the main configuration file contains errors.

In the message text:

**statement**
The name of the configuration statement that contains errors.

System action
Policy Agent processing continues. The specific action that Policy Agent takes depends on the configuration statement, as follows:

**AutoMonitorApps**
Automatic application monitoring does not occur.

**AutoMonitorParms**
Automatic application monitoring takes place using default values.

Operator response
Contact the system programmer. If the system programmer indicates that more information is required in the log file, restart the Policy Agent with a minimum of LogLevel 127 configured in the configuration file, or with the -d 1 start option.

System programmer response
Examine the log file to determine the cause of the problem. Correct the Policy Agent configuration errors identified in the log and restart the Policy Agent. If you need more information to diagnose the errors, re-create the error with a minimum of LogLevel 127 or start the Policy Agent with the -d 1 start option to generate the configuration errors. See the information about policy configuration files in z/OS Communications Server: IP Configuration Reference for the syntax of the failing statement.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

Module
pinit

Routing code
10

Descriptor code
12

Automation
Not applicable.
EZD1585I PAGENT AUTOMONITORAPPS STATEMENT CONTAINS ERRORS

EZD1586I  PAGENT HAS INSTALLED ALL LOCAL POLICIES FOR image

Explanation
Policy Agent finished installing all locally defined policies that were successfully processed in a configured TCP/IP stack. This message is issued only during Policy Agent initialization; it is not issued for the MODIFY UPDATE or REFRESH commands or for any policy updates that are detected after initialization has completed.

If there are errors in some policies that prevent those policy types from being installed, then this message indicates only that the policy types that were successfully processed have been installed. If errors prevent all configured policy types from being installed, this message is still issued but no policies are installed. You should always check for message EZZ8438I which indicates that policy errors have been detected, and take appropriate actions to correct those errors.

In the message text:

image
The name of the TCP/IP stack.

System action
Policy Agent processing continues.

Operator response
Not applicable.

System programmer response
Not applicable.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

Module
plfmmisc

Routing code
10

Descriptor code
12
Automation
This message is written to the system console and syslog. This message is a good candidate for automation. Automation can alert you to when Policy Agent has installed all locally defined policies, so that applications that use those policies can be started.

Example

EZD1586I PAGENT HAS INSTALLED ALL LOCAL POLICIES FOR TCPIP1

EZD1587I PAGENT IS UNABLE TO MONITOR APPLICATIONS

Explanation
Policy Agent is unable to monitor applications to determine whether they are active or inactive, or to start, restart, or stop the applications. Possible reasons for this error include configuration errors on the AutoMonitorApps or AutoMonitorParms statement, or an internal error.

System action
Policy Agent processing continues but the automatic application monitoring function is not available.

Operator response
Contact the system programmer. If the system programmer indicates that more information is required in the log file, restart the Policy Agent with a minimum of LogLevel 127 configured in the configuration file, or with the -d 1 start option.

System programmer response
Examine the log file to determine the cause of the problem. If message EZD1585I was issued, correct the Policy Agent configuration errors identified in the log and restart the Policy Agent. If you need more information to diagnose the errors, re-create the error with a minimum of LogLevel 127 or start the Policy Agent with the -d 1 start option. See the information about the AutoMonitorApps statement and the AutoMonitorParms statement in z/OS Communications Server: IP Configuration Reference for the syntax of the statements. See the information about diagnosing Policy Agent problems in z/OS Communications Server: IP Diagnosis Guide for information about gathering documentation for internal errors.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

Module
pmonapps

Routing code
10

Descriptor code
12

1030 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**EZD1588I  PAGENT MONITOR INFORMATION**

**Explanation**
A MODIFY procname,MON,DISPLAY command was issued. This message is followed by information about the set of applications that are eligible for automatic monitoring by the Policy Agent.

**System action**
Policy Agent processing continues.

**Operator response**
Not applicable.

**System programmer response**
Not applicable.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

**Module**
pzosinit

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**
Not applicable.

**EZD1589I  PAGENT HAS INSTALLED ALL REMOTE POLICIES FOR image**
Explanation
Policy Agent finished installing all policies that are defined on a remote policy server and were successfully processed in a configured TCP/IP stack. This message is issued only during Policy Agent initialization; it is not issued for the MODIFY UPDATE or REFRESH commands, or for any policy updates that are detected after initialization has completed.

If there are errors in some policies that prevent those policy types from being installed, then this message indicates only that the policy types that were successfully processed have been installed. If errors prevent all configured policy types from being installed, this message is still issued but no policies are installed. You should always check for message EZZ8438I which indicates that policy errors have been detected, and take appropriate actions to correct those errors.

In the message text:

image
The name of the TCP/IP stack.

System action
Policy Agent processing continues.

Operator response
Not applicable.

System programmer response
Not applicable.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

Module
plfmmisc

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and syslog. This message is a good candidate for automation. Automation can alert you to when Policy Agent has installed all remotely defined policies, so that applications that use those policies can be started.
EZD1589I PAGENT HAS INSTALLED ALL REMOTE POLICIES FOR TCPIP1

EZD1590I PAGENT UPDATE NOTIFICATION FAILED WITH RETURN CODE errno REASON CODE errnojr

Explanation
This message is received when one of the following types of Policy Agent update notification fails:

- Stack startup notification for every stack defined on the TcpImage statement. See TcpImage and PEPInstance statements in z/OS Communications Server: IP Configuration Reference for details.
- If the Policy Agent is started with the -i/I option, update notification in real time for changes to local files (all configuration files). See starting Policy Agent from the z/OS shell in z/OS Communications Server: IP Configuration Reference for details.

errno is the z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

errnojr is the hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing continues without update notification with the following results:

- If the Policy Agent is not notified about starting stacks, the stacks do not have their required policies.
- If the Policy Agent is not notified about updates to local configuration files, new policies are not updated in real time.

Operator response
Contact the system programmer.

System programmer response
Look up the return code and reason code values for the proper action to take. Also examine the Policy Agent log file to determine the name of the files that are having the problem. The following example shows what to look for in the Policy Agent log file:

```plaintext
SYSERR :004: plfm_update_event_register: Error Registering file '/tmp/tcpname.Pagent.tmp', IPC msg type = 9, rc=-1, errno (247) = EDC5247I Operation not supported., errno2=11800631
```

Correct the problem and instruct the operator to restart the Policy Agent if you want update notification.

When the registering file is of the form '/tmp/tcpname.Pagent.tmp' and the file is located in a read/write sysplex-aware zFS file system, a symbolic link can be created to a file in a z/OS UNIX file system or in a read-only zFS file system.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)
Module
plfmmisc

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and syslog for the Policy Agent. You might want to automate on this message to determine when the Policy Agent is not providing update notification and take corrective action.

Example

```
$HASP100 PAGENT   ON STCINRDR
IEF695I START PAGENT   WITH JOBNAME PAGENT   IS ASSIGNED TO USER IBMUSER,
GROUP SYS1
$HASP373 PAGENT   STARTED
IEF403I PAGENT - STARTED - TIME=09.54.14
EZZ8431I PAGENT STARTING
EZZ8432I PAGENT INITIALIZATION COMPLETE
EZD1590I PAGENT UPDATE NOTIFICATION FAILED WITH RETURN CODE 247 REASON CODE 11800631
```

```
EZD1591I CONNECTION TO POLICY SERVER FAILED DUE TO DUPLICATE CLIENT NAME clientName
```

Explanation
The attempt to establish a connection between the Policy Agent that is acting as a policy client and the Policy Agent that is acting as a policy server failed because of duplicate client names on the policy server. For the policy client for each TCP/IP stack, the PolicyServer statement provides the processing and security information for the policy server, including the client's name. If you have configured multiple clients to connect to a policy server you need to verify that each client has a unique name by checking the PolicyServer statement ClientName parameter.

In the message text:

```
clientName
```

The duplicate client name of the client that attempted to connect to a policy server.

System action
The policy server continues processing. The policy client uses the configured connection-wait parameter and connection-retry parameter on the ServerConnection statement to automatically retry the primary and backup connections until a connection is established. The policy client can retrieve the configured remote policies only after the connection is established.

Operator response
Save the system log for problem determination and contact the system programmer.

System programmer response
Verify that the client name for each policy client that connects to the policy server is unique. The client name is specified on the ClientName parameter of the PolicyServer statement for each TCP/IP stack for which policy is being retrieved. See "Steps for configuring the Policy Agent" in z/OS Communications Server: IP Configuration Guide for information on setting up the policy server and client configurations. Correct any configuration errors and restart the Policy Agent on the system where the configuration changes were made.
If needed, examine the log files to determine the duplicate client name that prevented a connection between the policy client and the policy server. If you need more information to diagnose the errors, re-create the error with a minimum of LogLevel 127 and start the policy server or policy client with the -d 128 start option.

**User response**
No action is needed.

**Problem determination**
See the System Programmer Response.

**Source**
z/OS Communications Server TCP/IP: Policy Agent (PAGENT)

**Module**
paapi.c

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to the system console and syslog for the Policy Agent. You might want to automate on this message to determine when a client connection fails due to a duplicate name.

**Example**

<table>
<thead>
<tr>
<th>EZD1591I CONNECTION TO POLICY SERVER FAILED DUE TO DUPLICATE CLIENT NAME CLIENT1</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1601I INTERNAL ERROR IN MODULE mod_id : err_id</td>
</tr>
</tbody>
</table>

**Explanation**
The Defense Manager daemon (DMD) detected an internal error. Additional diagnostic messages might be issued.

In the message text:

- **mod_id**
  An internal identifier that indicates the module that detected the error.

- **err_id**
  An internal identifier for this error in the detecting module.

- **value1**
  Internal error information.

- **value2**
  Internal error information.

- **value3**
  Internal error information.
System action
Results are unpredictable. One or more address space dumps might be produced with dump titles that match the message text.

Operator response
Contact the system programmer.

System programmer response
Contact IBM software support services and provide the syslog that includes this message. If available, provide CTRACE information for component SYSTCPDM and any dumps associated with this message.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1601I INTERNAL ERROR IN MODULE EZADMSSP : 0001 | 0 | 0 | 0

EZD1602I Internal Error err_id in module modname - unable to obtain memory of size size

Explanation
The Defense Manager daemon (DMD) was unable to obtain the required amount of memory.

In the message text:

err_id
The code that helps IBM service representatives identify the specific memory allocation request.

modname
The name of the module that encountered the error.

size
The amount of memory requested, in bytes.
**System action**
The current operation fails and the DMD attempts to continue processing.

**Operator response**
Free some memory and try the operation again. If the operation continues to fail, contact the system programmer with the message information.

**System programmer response**
See the information about diagnosing storage abends and storage growth in *z/OS Communications Server: IP Diagnosis Guide* for more information about storage problems. If you cannot resolve the memory allocation shortage, contact IBM software support services and provide the *modname* value and the *err_id* value.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

```
EZD1602I Internal Error 0001 in module EZADMMTH - unable to obtain memory of size 288
```

```
EZD1603I THE DEFENSE MANAGER DAEMON FAILED TO WRITE ITS PROCESS ID pid TO filename - ERRNO errno ERRNO DESCRIPTION description
```

**Explanation**
A file system error occurred when the Defense Manager daemon (DMD) tried to write its process ID to a file.

In the message text:

- **pid**
  - The DMD process ID.

- **filename**
  - The file into which the DMD was trying to write its process ID.
**errno**
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

**description**
Describes the meaning of the errno value.

**System action**
Defense Manager daemon (DMD) processing continues without a record of the process ID written to the file system.

**Operator response**
This error is not a problem unless the process ID file will be used for automation (for example, to automate the process of ending the DMD). Contact the System Programmer if the process ID file will be used for automation.

**System programmer response**
This error is not a problem unless the process ID file will be used for automation (for example, to automate the process of ending the DMD). Use the error information in the message to resolve the problem and restart the DMD. See the information about configuring the DMD in z/OS Communications Server: IP Configuration Guide for information about setting the pid file location.

The directory for the process ID file must exist and the DMD user ID must have authority to write to the directory.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**
```
EZD1603I  THE  DEFENSE  MANAGER  DAEMON  FAILED  TO  WRITE  ITS  PROCESS  ID  19  TO  /tmp-baddir/dmd.pid  -  ERRNO  129
ERRNO  DESCRIPTION  EDC5129I  No  such  file  or  directory.
```

EZD1604I  THE  DEFENSE  MANAGER  DAEMON  SHUTDOWN  SEQUENCE  HAS  BEGUN
**Explanation**
The Defense Manager daemon (DMD) has begun its shutdown sequence.

**System action**
The daemon is in the process of shutting down. No more work is processed.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

<table>
<thead>
<tr>
<th>EZD1604I</th>
<th>THE DEFENSE MANAGER DAEMON SHUTDOWN SEQUENCE HAS BEGUN</th>
</tr>
</thead>
</table>

| EZD1605I | THE DEFENSE MANAGER DAEMON SHUTDOWN SEQUENCE HAS COMPLETED |

**Explanation**
The Defense Manager daemon (DMD) has completed its shutdown sequence.

**System action**
The DMD ends.

**Operator response**
None.
**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

```
EZD1605I THE DEFENSE MANAGER DAEMON SHUTDOWN SEQUENCE HAS COMPLETED

EZD1606I The ExcludeAddress value value on line linenum is not valid - reason
```

**Explanation**
A value that is not valid was detected for the ExcludeAddress keyword in the Defense Manager daemon (DMD) configuration file.

In the message text:

- **value**
  The value that is not valid.

- **linenum**
  The line number in the DMD configuration file where the value occurs.

- **reason**
  The reason the value is not valid.

**System action**
The processing of the configuration file stops. If this error occurred during the processing of a MODIFY command, then no changes are committed and the daemon continues using the old configuration values. If this error occurred during the initial startup sequence of the daemon, then the daemon ends.

**Operator response**
Contact the system programmer.
System programmer response
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Correct the ExcludeAddress value that is in error.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1606I The ExcludeAddress value 10.1.1.1.1 on line 156 is not valid - ipaddress is not valid

EZD1607I The ExcludeAddress value value on line linenum cannot be accepted because the limit of limit exclusion addresses for a single stack has already been reached

Explanation
The number of ExcludeAddress keywords that can be specified for a DmStackConfig statement is limited. More ExcludeAddress keywords were specified for a DmStackConfig statement in the Defense Manager daemon (DMD) configuration file than is allowed.

In the message text:
value
The ExcludeAddress value that cannot be accepted.

linenum
The line number in the DMD configuration file where ExcludeAddress keyword occurs.

limit
The number of ExcludeAddress keywords allowed on a DmStackConfig statement.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.
**Operator response**
Contact the system programmer.

**System programmer response**
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Update the DmStackConfig statement so that it does not specify too many ExcludeAddr keywords.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

```
EZD1607I The ExcludeAddress value 10.7.7.7 on line 163 cannot be accepted because the limit of 10 exclusion addresses for a single stack has already been reached

EZD1608I THE DEFENSE MANAGER DAEMON FAILED TO START BECAUSE THE DAEMON IS ALREADY RUNNING - TOKEN tokenname LEVEL level PERSIST persist RETURN CODE retcode
```

**Explanation**
The Defense Manager daemon (DMD) started and determined that another instance of the DMD is already running. Only one DMD can be running at a time.

In the message text:

- **tokenname**
  The name of the MVS token to which the server is trying to get exclusive access. This value is DEFENSE MANAGER for the DMD.

- **level**
  The level of exclusivity required. This value is 4 for the DMD.

- **persist**
  Possible values are:
The token is released when the server stops. This is the value for the DMD.

The token persists after the server ends.

redo
code
The return code from the MVS token service. See z/OS MVS Programming: Authorized Assembler Services Reference EDT-IXG for a complete list of IEANTCR return and reason codes.

System action
This instance of the DMD ends.

Operator response
If you want to start a new instance of the DMD, stop the active DMD first.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

```
EZD1608I THE DEFENSE MANAGER DAEMON FAILED TO START BECAUSE THE DAEMON IS ALREADY RUNNING - TOKEN DEFENSE MANAGER LEVEL 4 PERSIST 0 RETURN CODE 4

EZD1609I DEFENSE MANAGER DAEMON RELEASE release SERVICE LEVEL level CREATED ON date
```

Explaination
This message is the first message printed to the console when the Defense Manager daemon (DMD) is started.

In the message text:
**release**
The release name.

**level**
The service level name.

**date**
The build date of the daemon.

**System action**
None.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

```
EZD1609I  DEFENSE MANAGER DAEMON RELEASE CS V1R10 SERVICE LEVEL CS070815  CREATED ON Aug 15 2007
```

**EZD1610I**
THE DEFENSE MANAGER DAEMON INITIALIZATION SEQUENCE HAS BEGUN

**Explanation**
This message is a notification that the Defense Manager daemon (DMD) has begun its initialization sequence.

**System action**
The Defense Manager daemon (DMD) is starting.
Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1610I THE DEFENSE MANAGER DAEMON INITIALIZATION SEQUENCE HAS BEGUN

EZD1611I THE DEFENSE MANAGER DAEMON INITIALIZATION SEQUENCE HAS COMPLETED

Explanation
This message is a notification that the Defense Manager daemon (DMD) has completed its initialization sequence.

System action
The DMD is ready.

Operator response
None.

System programmer response
None.
User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1611I THE DEFENSE MANAGER DAEMON INITIALIZATION SEQUENCE HAS COMPLETED

EZD1612I ERROR (errno | errnojr | description) WHILE OPENING MESSAGE CATALOG name - DEFAULT MESSAGES WILL BE USED

Explanation
The Defense Manager daemon (DMD) was unable to open the message catalog. Default messages will be used.

In the message text:

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

errnojr
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

description
Describes the meaning of the errno value.

name
The message catalog file that the DMD was attempting to open.

System action
DMD processing continues. DMD will use the internal default messages instead of the messages from the external message catalog.
Operator response
If the default messages are acceptable, no action is necessary. Otherwise, contact the system programmer.

System programmer response
Correct the error indicated by the `errno`, `errnojr`, and `description` values. There are several possible causes of this error, such as file or directory permissions that do not allow read access. See z/OS XL C/C++ Runtime Library Reference for more information about the `catopen()` function call. A common cause of this error message is an incorrectly set NLSPATH environment variable. See the information about the NLSPATH environment variable in z/OS UNIX System Services Programming Tools.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
```plaintext
EZD1612I ERROR ( 129 | C90F001E | EDCS129I No such file or directory. ) WHILE OPENING MESSAGE CATALOG dmdmsg.cat - DEFAULT MESSAGES WILL BE USED
```

A message-generated dump has been created with the title `title`

Explanation
A Defense Manager daemon (DMD) syslog message generated an address space dump.

In the message text:

`title`

The text associated with the dump. The title contains the message number and associated message text that caused the dump to be generated.

System action
After the dump is created, the DMD continues processing.
Operator response
Contact the system programmer.

System programmer response
Obtain the system log and the generated dump, and contact IBM software support.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1613I A message-generated dump has been created with title DMD Message generated dump EZD1601I
INTERNAL ERROR IN MODULE EZADMMTH 0015 - 0 | 0 | 0

EZD1614I A message-generated dump was suppressed for message message

Explanation
A Defense Manager daemon (DMD) syslog message attempted to generate an address space dump; however, no more than two message-generated dumps can be created in a 15-minute period, so the dump was suppressed.

In the message text:
message
The message that attempted to generate the dump.

System action
DMD processing continues.

Operator response
Contact the system programmer.
System programmer response
Capture the system log and any message-generated dumps that were created earlier. Contact IBM software support services to analyze this data.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
```
EZD1614I A message-generated dump was suppressed for message EZD1601I INTERNAL ERROR IN MODULE EZADMMTH 0015 - 0 | 0 | 0

EZD1615I INCORRECT SYNTAX FOR THE FILE= PORTION OF THE DEFENSE MANAGER DAEMON MODIFY COMMAND (specification)
```

Explanation
The file specification on a MODIFY procname,REFRESH,FILE=file command was incorrect. The file name must be a fully qualified z/OS UNIX file name or an MVS data set name. A z/OS UNIX file name must be enclosed by single quotation marks. MVS data set names must begin with two forward slashes and the data set name must be enclosed by single quotation marks.

In the message text:

specification
  The incorrect FILE= specification.

System action
The MODIFY command is ignored. Processing continues using the previous configuration values.

Operator response
Correct the file name specification and issue the command again. See the information about the Defense Manager daemon MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information about the syntax of the file name specification.
**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**
In the following example, the z/OS UNIX file name was not enclosed by single quotation marks which causes the error.

```
EZD1615I INCORRECT SYNTAX FOR THE FILE= PORTION OF THE DEFENSE MANAGER DAEMON MODIFY COMMAND ( ,FILE=/U/USER1/DMD.CONF )
```

In the following example, FILE was misspelled.

```
EZD1615I INCORRECT SYNTAX FOR THE FILE= PORTION OF THE DEFENSE MANAGER DAEMON MODIFY COMMAND ( ,FILE='/etc/security/dmd.conf' )
```

**Explanation**
An error occurred during the processing of a Defense Manager daemon (DMD) configuration file in response to a MODIFY REFRESH command.

In the message text:

- **filename**
  The name of the configuration file that was being processed.

- **rc**
  The return code. The possible return codes are:
  - 2
    The file does not exist or could not be opened.
An error was detected while processing one of the statements in the file.

**System action**

Processing of the configuration file stops. No changes are committed and the DMD continues using the old configuration values.

**Operator response**

If the rc value is 2, check that the name of the configuration was specified correctly. Reissue the MODIFY command using the correct file name. If problems persist or if the rc value is 3, contact the system programmer.

**System programmer response**

Verify that the configuration file exists and that the DMD has permission to read it. If the rc value is 3, check for a more specific configuration error message in the system log and correct the DMD configuration file error.

See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**

Not applicable.

**Routing code**

10

**Descriptor code**

12

**Automation**

Not applicable.

**Example**

```
EZD1616I AN ERROR OCCURRED WHILE READING THE DEFENSE MANAGER DAEMON CONFIGURATION FILE /u/user1/dmd.conf - RETURN CODE 2 - MODIFY REFRESH COMMAND IS REJECTED

EZD1617I AN ERROR OCCURRED WHILE READING THE DEFENSE MANAGER DAEMON CONFIGURATION FILE filename - RETURN CODE rc
```
Explanation
An error occurred while processing the Defense Manager daemon (DMD) configuration file during the initial startup sequence. Additional messages will be issued to provide more specific information.

In the message text:

filename
The name of the configuration file that is being processed.

rc
The return code. Possible values are:

2
The file does not exist or could not be opened.

3
An error was detected while processing one of the statements in the file.

System action
DMD initial startup processing fails and the DMD ends.

Operator response
Contact the system programmer.

System programmer response
Verify that the configuration file exists and that the DMD has permission to read it. If the rc value is 3, check for a more specific configuration error message in the system log and correct the DMD configuration file error.

See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.
Example
EZD1617I AN ERROR OCCURRED WHILE READING THE DEFENSE MANAGER DAEMON CONFIGURATION FILE /etc/security/dmd.conf - RETURN CODE 3

EZD1618I Configuration keyword keyword on line linenum requires a value

Explanation
The keyword requires a value but a value was not specified in the Defense Manager daemon (DMD) configuration file.

In the message text:
keyword
   The name of the keyword that is missing a value.
linenum
   The line number in the DMD configuration file of the keyword that requires a value.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the previous configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
Correct the error in the configuration file and restart the DMD or reissue the MODIFY command.
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12
Automation
Not applicable.

Example

EZD1618I Configuration keyword SyslogLevel on line 74 requires a value

EZD1619I The configuration value value for keyword keyword on line linenum contains one or more characters that are not allowed

Explanation
The Defense Manager daemon (DMD) was processing a configuration file and the keyword value contained one or more characters that are not allowed.

In the message text:

value  The keyword value that contains one or more characters that are not allowed.
keyword  The keyword with the incorrect value.
linenum  The line number in the DMD configuration file where the incorrect keyword value occurs.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
Correct the error in the configuration file and restart the DMD or reissue the MODIFY command.
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10
Descriptor code
12

Automation
Not applicable.

Example

EZD1619I The configuration value FF for keyword SyslogLevel on line 75 contains one or more characters that are not allowed

EZD1620I The keyword value value on line linenumber is outside of the allowable range lowvalue - highvalue

Explanation
The value configured for the specified Defense Manager daemon (DMD) keyword is outside the allowable range.

In the message text:

**keyword**
The name of the DMD configuration keyword.

**value**
The value configured for the keyword in the configuration file

**linenum**
The line number where the error occurred in the configuration file.

**lowvalue**
The lowest value allowed for the keyword.

**highvalue**
The highest value allowed for the keyword.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
Correct the error in the configuration file and restart the DMD or reissue the MODIFY command.

See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon
Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1620I The SyslogLevel value 256 on line 75 is outside of the allowable range of 0 - 255

EZD1621I AN ERROR OCCURRED WHILE TRYING TO ACCESS DEFENSIVE FILTER DIRECTORY dirname - ERRNO errno description

Explanation
An error occurred when the Defense Manager daemon (DMD) tried to access the defensive filter directory.

In the message text:

dirname
The defensive filter directory that the DMD was trying to access.

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

description
Describes the meaning of the errno value.

System action
If this error occurs during the initial startup sequence of the daemon, the daemon ends. Message EZD1611I indicates the completion of the startup sequence. If this error occurs during the operation of the daemon, DMD processing will continue without access to the directory. If the DMD stops and is restarted it will no longer have any access to previously created defensive filters. The DMD must be able to write to the directory so that defensive filters persist when the DMD stops.

Operator response
Contact the system programmer.

System programmer response
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DefensiveFilterDirectory keyword in the DMD configuration file.

Ensure that the defensive filter directory exists, that it is on an undamaged file system, and that the DMD user ID has the authority to create, delete, and write to files in the defensive filter directory.

If this error occurs during the DMD initial startup sequence, restart the DMD after you correct the error condition. If this error occurs during operation of the daemon, attempt to correct the error without stopping the DMD. After
you correct the error, the DMD will be able to update its persistent filter storage that is preventing any loss of unexpired filters. The DMD does not need to be restarted after correcting the error condition.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

```
EZD1621I AN ERROR OCCURRED WHILE TRYING TO ACCESS DEFENSIVE FILTER DIRECTORY /var/dm_filters_alt2 - ERRNO 129 EDC5129I No such file or directory
```

EZD1622I DEFENSE MANAGER DAEMON CONFIGURATION PROCESSING IS COMPLETE USING FILE filename

Explanation
The Defense Manager daemon (DMD) successfully completed processing the configuration file. This message can follow the successful startup of the daemon or a successful MODIFY REFRESH command.

In the message text:

filename
The z/OS UNIX file name or MVS data set name that contains DMD configuration statements.

System action
None.

Operator response
None.

System programmer response
None.
User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1622I DEFENSE MANAGER DAEMON CONFIGURATION PROCESSING IS COMPLETE USING FILE /etc/security/dmd.conf

EZD1623I DEFENSE MANAGER DAEMON CONFIGURATION FILE NOT SPECIFIED AND DEFAULT CONFIGURATION FILE filename MISSING OR NOT VALID - USING DEFAULTS FOR CONFIGURATION PARAMETERS

Explanation
No configuration file was specified, so the Defense Manager daemon (DMD) attempted to use the default configuration file. The default configuration file does not exist or cannot be read so the DMD uses the default configuration values.

In the message text:
filename
The default configuration file name.

System action
DMD processing continues using the default values for configuration parameters.

Operator response
Contact the system programmer.

System programmer response
When there is not a configuration file, defensive filtering is inactive for all TCP/IP stacks on your system. The system-supplied defaults do not include any DmStackConfig statements. If you want values other than the system-supplied defaults, create a configuration file and activate it using the MODIFY REFRESH command.
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

<table>
<thead>
<tr>
<th>EZD1623I</th>
<th>DEFENSE MANAGER DAEMON CONFIGURATION FILE NOT SPECIFIED AND DEFAULT CONFIGURATION FILE /etc/security/dmd.conf MISSING OR NOT VALID - USING DEFAULTS FOR CONFIGURATION PARAMETERS</th>
</tr>
</thead>
</table>

**EZD1624I**  The Defense Manager daemon socket directory directory does not exist and cannot be created - errno errno description

**Explanation**
The Defense Manager daemon (DMD) did not find a preexisting socket directory and was unable to create the directory.

In the message text:

- **directory**  The socket directory that does not exist and that cannot be created by DMD.
- **errno**  The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errno) information in z/OS UNIX System Services Messages and Codes.
- **description**  Describes the meaning of the errno value.

**System action**
DMD initialization stops and the daemon ends.

**Operator response**
Contact the system programmer.
System programmer response

Ensure that the socket directory exists or that the DMD has the correct permissions to create it. If the DMD needs to create the socket directory, all elements of the directory path must exist, except for the last element. For example, if the socket directory identified in the message is /var/sock, then at least the /var directory must exist. If the DMD is defined with a nonzero UID, create the /var/dm directory before starting the DMD. The directory should be owned by the DMD user ID and the DMD should be able to create, delete, read, and write files to the directory.

User response

Not applicable.

Problem determination

None.

Source

z/OS Communications Server TCP/IP: Defense Manager daemon

Module

Not applicable.

Routing code

10

Descriptor code

12

Automation

Not applicable.

Example

EDC5129I no such file or directory.
EZD1624I The Defense Manager daemon socket directory /var/dm does not exist and cannot be created - errno 111.
EDC5111I Permission denied.

EZD1625I THE DEFENSE MANAGER DAEMON COULD NOT INITIALIZE MODIFY AND STOP COMMAND SUPPORT

Explanation

The Defense Manager daemon (DMD) was unable to initialize the MODIFY and STOP command support.

System action

DMD processing continues. The MODIFY and STOP command service is not available.

Operator response

If you require the MODIFY or STOP command service, restart the DMD. If the problem persists, contact the system programmer.
System programmer response
Contact IBM software support services and provide the syslog that includes this message. If available, provide CTRACE information for component SYSTCPDM.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1625I THE DEFENSE MANAGER DAEMON COULD NOT INITIALIZE MODIFY AND STOP COMMAND SUPPORT

EZD1626I THE DEFENSE MANAGER DAEMON CTRACE PARMLIB MEMBER pname WAS NOT FOUND

Explanation
The Defense Manager daemon (DMD) was unable to find the specified parmlib member and was initialized with the MINIMUM tracing option.

In the message text:

pname
The name of the CTRACE parmlib member.

System action
DMD CTRACE initializes with the MINIMUM tracing option; the DMD continues processing.

Operator response
If different CTRACE options are required, contact the system programmer.

System programmer response
If different CTRACE options are required, configure the CTRACE parmlib member.
See the information about TCP/IP services component trace for DMD in z/OS Communications Server: IP Diagnosis Guide for more information about configuring the CTRACE parmlib member and enabling CTRACE after DMD initialization.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

```
EZD1626I THE DEFENSE MANAGER DAEMON CTRACE PARMLIB MEMBER CTIDMD00 WAS NOT FOUND

EZD1627I SYNTAX ERROR IN THE DEFENSE MANAGER DAEMON CTRACE PARMLIB MEMBER *pname*
```

**Explanation**
A syntax error was detected in the specified parmlib member, which is used to configure the Defense Manager daemon (DMD) CTRACE options. The DMD CTRACE is initialized with the MINIMUM tracing option.

In the message text:

`*pname`

The name of the CTRACE parmlib member.

**System action**
The DMD CTRACE initializes with the MINIMUM tracing option; the DMD continues processing.

**Operator response**
Contact the system programmer.

**System programmer response**
Correct the syntax error in the parmlib member.
See the information about TCP/IP services component trace for DMD in z/OS Communications Server: IP Diagnosis Guide for more information about configuring the CTRACE parmlib member and enabling CTRACE after DMD initialization.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

```
EZD1627I SYNTAX ERROR IN THE DEFENSE MANAGER DAEMON CTRACE PARMLIB MEMBER CTIDMD00
EZD1628I DEFENSE MANAGER DAEMON CTRACE INITIALIZATION ERROR - FUNCTION function RETURN CODE rc REASON CODE rsn
```

**Explanation**
The Defense Manager daemon (DMD) failed to initialize the CTRACE subsystem.

In the message text:

*function*
  The function that was being processed when the CTRACE error occurred.

*rc*
  The error return code.

*rsn*
  The error reason code.

**System action**
The DMD continues processing without CTRACE enabled.

**Operator response**
See the information about defining a user application to the component trace service in z/OS MVS Programming: Authorized Assembler Services Reference ALE-DYN for the return code and reason code explanations for the different CTRACE functions.
See the information about TCP/IP services component trace for the DMD in z/OS Communications Server: IP Diagnosis Guide for more information about configuring the CTRACE parmlib member and enabling CTRACE after DMD initialization.

Ensure that storage is available for the size of the trace buffers. If these checks do not reveal the cause of the problem, contact the system programmer. If you want the DMD to be enabled for CTRACE, stop and restart the DMD when the cause of the problem is identified and resolved.

System programmer response
If the problem cannot be resolved, contact IBM software support services and provide the syslog that includes this message.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

```
EZD1628I DEFENSE MANAGER DAEMON CTRACE INITIALIZATION ERROR - FUNCTION CTSSM RETURN CODE 00000004 REASON CODE 00000004
```

**EZD1629I**  Keyword or value rejected - **Keyword** keyword **Value** value

Explanation
The Defense Manager daemon (DMD) was processing a configuration file and either the keyword is unsupported or the value is not valid for the keyword.

In the message text:

- **keyword**  The keyword portion of the configuration line.
- **value**  The value entered for the keyword. If a value was not entered for the keyword, this field is not displayed.
System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
Check for a more specific configuration error message preceding this one. See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Correct the keyword or value that is in error.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1629I Keyword or value rejected - Keyword SyslogLevel Value FF

EZD1630I Right brace (}) expected, but not found

Explanation
The Defense Manager daemon (DMD) was processing a configuration file and the right brace (}) of a configuration statement was missing. All statements begin with a left brace ({) and end with a right brace (}). Each brace must be on a separate line with no other text.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.
Operator response
Contact the system programmer.

System programmer response
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Correct the error in the configuration file and restart DMD or reissue the MODIFY command.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1630I Right brace (}) expected, but not found

EZD1631I statementname is not a recognized statement type

Explanation
An unrecognized statement type is in the Defense Manager daemon (DMD) configuration file.
In the message text:
statementname
The name of the unrecognized statement.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.
System programmer response
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Correct the error in the configuration file and restart DMD or reissue the MODIFY command.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1631I DmStkConfig is not a recognized statement type

EZD1632I Left brace (:) expected, but not found

Explanation
The left brace (:) of a configuration statement was missing. All statements begin with a left brace (:) and end with a right brace (:). Each brace must be on a separate line with no other text.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Correct the error in the configuration file and restart DMD or reissue the MODIFY command.
User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1632I Left brace ( { ) expected, but not found
EZD1633I THE MODIFY COMMAND EXCEEDED THE MAXIMUM ALLOWED LENGTH OF maxlen

Explanation
The Defense Manager daemon (DMD) MODIFY command can accept only the first maxlen characters entered.

In the message text:

maxlen
The maximum number of characters that the MODIFY command can accept.

System action
The MODIFY command is ignored.

Operator response
Reissue the MODIFY command without exceeding the length limit.

System programmer response
None.

User response
Not applicable.

Problem determination
None.
Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>EZD1633I</th>
<th>THE MODIFY COMMAND EXCEEDED THE MAXIMUM ALLOWED LENGTH OF 128</th>
</tr>
</thead>
</table>

**EZD1634I** THE COMMAND ENTERED IS NOT A RECOGNIZED MODIFY REQUEST - *input*

Explanation
You entered an unsupported MODIFY request.

In the message text:

*input*

- The MODIFY command that is not supported.

System action
The MODIFY command is ignored.

Operator response
Correct and reissue the MODIFY command. See the information about the Defense Manager daemon MODIFY command in z/OS Communications Server: IP System Administrator's Commands for more information about the MODIFY command.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon
Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
THE COMMAND ENTERED IS NOT A RECOGNIZED MODIFY REQUEST - this is the bad input

EZD1635I THE MODIFY SUBCOMMAND IS NOT SUPPORTED BY THE DEFENSE MANAGER DAEMON - subcommand

Explanation
You entered a MODIFY command that included the specified subcommand. This MODIFY subcommand is not supported by the Defense Manager daemon (DMD).

In the message text:
subcommand
  The MODIFY subcommand that is not supported by the DMD.

System action
The MODIFY command is ignored.

Operator response
Correct and reissue the MODIFY command. See the information about the Defense Manager daemon MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information about the MODIFY command.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

```
EZD1635I THE MODIFY SUBCOMMAND IS NOT SUPPORTED BY THE DEFENSE MANAGER DAEMON - REFRESH
```

```
EZD1636I THE DEFENSE MANAGER DAEMON RECEIVED THE STOP COMMAND
```

Explanation

The Defense Manager daemon (DMD) recognizes that a STOP command was issued on the console.

System action

The DMD begins its shutdown sequence.

Operator response

None.

System programmer response

None.

User response

Not applicable.

Problem determination

None.

Source

z/OS Communications Server TCP/IP: Defense Manager daemon

Module

Not applicable.

Routing code

10

Descriptor code

12

Automation

Not applicable.
Example
EZD1636I THE DEFENSE MANAGER DAEMON RECEIVED THE STOP COMMAND

EZD1637I Exception classname encountered in module modname - error id internal_error_ID rc rc errno errnojr errnojr

Explanation
A runtime exception was issued.
In the message text:

classname
The name of the exception class.

modname
The name of the module in which the exception was detected.

internal_error_ID
An internal error ID used by IBM to identify the error detected in the module.

rc
The return code, if any, from the function call that caused or detected the error condition.

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errno) information in z/OS UNIX System Services Messages and Codes.

errnojr
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojr) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing continues in most cases. In some cases, the Defense Manager daemon (DMD) might experience problems or end.

Operator response
Contact the system programmer.

System programmer response
This message does not always indicate an unrecoverable DMD condition. However, any traces or logs that contain this message should be forwarded to IBM software support services if the daemon is unable to operate normally after these messages are logged.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1637I Exception ConstructorError encountered in module EZADMMTH - error id 1 rc 1 errno 0 errnojr 0

EZD1638I The stackname for the DmStackConfig statement on configuration file line linenum is missing or incorrect - stackname

Explanation
The stack name specified on a DmStackConfig statement in the Defense Manager daemon (DMD) configuration file is missing or incorrect.

In the message text:

- **linenum**
  - The line number in the DMD configuration file where the missing or incorrect stack name occurs.

- **stackname**
  - The incorrect stack name. If the stack name is missing, this field is not displayed.

System action
The processing of the configuration file stops. If this error occurs during the processing of a MODIFY command, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Correct the DmStackConfig statement with the incorrect or missing stack name. Restart DMD or reissue the MODIFY command.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon
Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>EZD1638I</th>
<th>The stackname for the DmStackConfig statement on configuration file line 157 is missing or incorrect - TCP S2</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1639I</td>
<td>THE DEFENSIVEFILTERDIRECTORY VALUE MAY NOT BE CHANGED BY A REFRESH OPERATION - IN USE initialdirectory FAILED refreshdirectory</td>
</tr>
</tbody>
</table>

Explanation
The Defense Manager daemon (DMD) configuration file used for the MODIFY REFRESH command contained a different DefensiveFilterDirectory value than the DMD configuration file with which DMD was started. If DefensiveFilterDirectory was not specified in the REFRESH configuration file the default value /var/dm/filters was used. The defensive filter directory cannot be changed with a MODIFY REFRESH command. Stop DMD, then restart DMD with the new defensive filter directory.

In the message text:

initialdirectory
The name of the defensive filter directory with which DMD was started.

refreshdirectory
The name of the defensive filter directory in the REFRESH configuration file that was rejected.

System action
Processing of the configuration file stops. No changes are committed and the daemon continues using the old configuration values.

Operator response
If the DMD should use a different defensive filter directory than the one with which it was started, stop and restart DMD with the configuration file containing the new defensive filter directory name.

System programmer response
None.

User response
Not applicable.

Problem determination
None.
Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1639I THE DEFENSIVEFILTERDIRECTORY VALUE MAY NOT BE CHANGED BY A REFRESH OPERATION - IN USE /var/dm/filters FAILED /var/dm/filters1

EZD1640I MISSING OR INCORRECT STACKNAME SPECIFIED FOR FORCE_INACTIVE MODIFY COMMAND - stackname

Explanation

The stack name was specified incorrectly on a MODIFY procname,FORCE_INACTIVE,stackname command.

In the message text:

stackname
The incorrect stack name. If the stack name is missing, this field is not displayed.

System action
The MODIFY command is ignored.

Operator response

Reissue the MODIFY command with the correct stack name. See the information about the Defense Manager daemon MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information about the MODIFY command.

System programmer response
None.

User response
Not applicable.

Problem determination
None

Source
z/OS Communications Server TCP/IP: Defense Manager daemon
Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
In this example, an incorrect stack name was specified for the MODIFY FORCE_INACTIVE command.

```
MODIFY DMD,FORCE_INACTIVE,TCP CS
```

The following message is the result:

```
EZD1640I MISSING OR INCORRECT SYNTAX FOR STACKNAME PORTION OF FORCE_INACTIVE MODIFY COMMAND - ,TCP CS
```

In this example, a stack name was not specified for the MODIFY FORCE_INACTIVE command.

```
MODIFY DMD,FORCE_INACTIVE
```

The following message is the result:

```
EZD1640I MISSING OR INVALID SYNTAX FOR STACKNAME PORTION OF FORCE_INACTIVE MODIFY COMMAND -
```

```
EZD1641I STACK stackname IS NOT STARTED AND IS NOT CONFIGURED IN THE DEFENSE MANAGER DAEMON CONFIGURATION FILE
```

Explanation
The MODIFY FORCE_INACTIVE command was issued for a TCP/IP stack that is not defined in the Defense Manager daemon (DMD) configuration file. The TCP/IP stack is also down. Defensive filtering is inactive for the specified TCP/IP stack so there is no action for DMD to take to force the defensive filter mode to inactive.

In the message text:

`stackname`
The name of the TCP/IP stack specified on the MODIFY FORCE_INACTIVE command.

System action
The MODIFY FORCE_INACTIVE command is ignored.

Operator response
None.

System programmer response
None.

User response
Not applicable.
Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1641I STACK STACK1 IS NOT STARTED AND IS NOT CONFIGURED IN THE DEFENSE MANAGER DAEMON CONFIGURATION FILE

EZD1642I INCORRECT SYNTAX ON THE MODIFY SUBCOMMAND subcommand

Explanation
The syntax of the MODIFY command is incorrect.
In the message text:
subcommand
  The subcommand portion of the MODIFY command.

System action
The MODIFY command is ignored.

Operator response
Correct and reissue the MODIFY command. See the information about the Defense Manager daemon MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information about the MODIFY command.

System programmer response
None.

User response
Not applicable.

Problem determination
None.
**Source**
z/OS Communications Server TCP/IP: Defense Manager daemon

**Module**
Not applicable.

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**
In this example, a stack name was incorrectly specified for the MODIFY DISPLAY command.

```
MODIFY  DMD,DISPLAY,TCPCS
```

The following message is the result:

```
EZD1642I INCORRECT SYNTAX ON THE MODIFY SUBCOMMAND DISPLAY
```

**Explanation**
A MODIFY FORCE_INACTIVE command was processed successfully, which forced the stack defensive filter mode to inactive.

In the message text:

*`stackname`*

The name of the TCP/IP stack specified on the MODIFY FORCE_INACTIVE command.

**System action**
Defensive filters are deleted from the TCP/IP stack and the defensive filter mode is set to inactive. The Defense Manager daemon (DMD) will not add any defensive filters to the TCP/IP stack while the mode is inactive.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.
Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1643I THE DEFENSIVE FILTER MODE FOR STACK TCPCS2 WAS SUCCESSFULLY FORCED TO INACTIVE

EZD1644I THE DEFENSIVE FILTER MODE FOR STACK stackname COULD NOT BE FORCED TO INACTIVE

Explanation
A MODIFY FORCE_INACTIVE command was not successful as the result of an internal error in the Defense Manager daemon (DMD) or TCP/IP stack.

In the message text:

stackname
The name of the TCP/IP stack on the MODIFY FORCE_INACTIVE command.

System action
Results are unpredictable. One or more address space dumps might be produced.

Operator response
Contact the system programmer.

System programmer response
Check the system log for related error messages. Contact IBM software support services and provide the syslog and any dumps.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon
Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1644I THE DEFENSIVE FILTER MODE FOR STACK TCPCS2 COULD NOT BE FORCED TO INACTIVE

EZD1645I There are multiple DmConfig statements in the configuration file - only one DmConfig statement is permitted

Explanation
Multiple DmConfig statements are specified in the Defense Manager daemon (DMD) configuration file. Only one DmConfig statement is allowed.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Correct the error in the configuration file and restart DMD or reissue the MODIFY command.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1645I There are multiple DmConfig statements in the configuration file - only one DmConfig statement is permitted

EZD1646I The keywordname keyword on configuration file line linenum is a duplicate - only one is permitted in each statementname statement

Explanation
A duplicate keyword occurs on the specified statement in the Defense Manager daemon (DMD) configuration file.

In the message text:

keywordname
  The name of the duplicate keyword.

linenum
  The line number in the DMD configuration file where the duplicate keyword occurs.

statementname
  The name of the statement in which the duplicate keyword occurs.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Correct the error in the configuration file and restart DMD or reissue the MODIFY command.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon
Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1646I The SyslogLevel keyword on configuration file line 76 is a duplicate - only one is permitted in each DmConfig statement

EZD1647I The TCP/IP stack name *stackname* was used on multiple DmStackConfig statements

Explanation
Multiple DmStackConfig statements specify the same TCP/IP stack in the Defense Manager daemon (DMD) configuration file.

In the message text:

*stackname*
The name of the TCP/IP stack that appears on multiple DmStackConfig statements.

System action
The processing of the configuration file stops. If this error occurs while a MODIFY command is being processed, then no changes are committed and the daemon continues using the old configuration values. If this error occurs during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Specify a unique TCP/IP stack name for each DmStackConfig statement. Restart the DMD or reissue the MODIFY command.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon
Module
Not applicable.

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1647I The TCP/IP stack name TCPCS was used on multiple DmStackConfig statements

EZD1648I DefensiveFilterDirectory cannot be a relative path - dirname

Explanation
The DefensiveFilterDirectory in the Defense Manager daemon (DMD) configuration file was specified as a relative path. The DefensiveFilterDirectory must be specified as an absolute path.

In the message text:

dirname
The name of the DefensiveFilterDirectory specified in the configuration file.

System action
DMD initial startup processing fails and the DMD ends.

Operator response
Contact the system programmer.

System programmer response
Correct the directory specified for DefensiveFilterDirectory in the DMD configuration file and restart the DMD. See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1648I DefensiveFilterDirectory cannot be a relative path - var/dm_filters

EZD1649I Extra text at end of configuration file line linenum is not permitted

Explanation
Extraneous text was found on a line in the Defense Manager daemon (DMD) configuration file.
In the message text:

*linenum*  
The line number in the DMD configuration file where the extraneous text was found.

System action
The processing of the configuration file stops. If this error occurred during the processing of a MODIFY command, then no changes are committed and the daemon continues using the old configuration values. If this error occurred during the initial startup sequence of the daemon, then the daemon ends.

Operator response
Contact the system programmer.

System programmer response
Extraneous text cannot be placed at the end of the line in the DMD configuration file. A comment must be specified on a separate line beginning with a number sign (#). All statements begin with an opening brace ( { ) and end with a closing brace ( } ). Each brace must be on a separate line with no other text. See the information about the Defense Manager daemon (DMD) in z/OS Communications Server: IP Configuration Reference for information about the DMD configuration file. Remove the extra text at the end of the specified line in the DMD configuration file. Restart DMD or reissue the MODIFY command.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1649I Extra text at end of configuration file line 75 is not permitted

EZD1650I Unable to open message catalog cat: errno errno (description) errnojr errnojr - Default messages will be used

Explanation
An attempt was made to open the nssctl command message catalog in the message catalog directory, but the catalog could not be opened for the specified reason. The default location for the message catalog is set by the NLSPATH environment variable.

In the message text:

cat  
The name of the catalog the nssctl command attempted to open.

errno  
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

description  
Describes the meaning of the errno value.

errnojr  
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (ernojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The nssctl command processing continues. Default messages will be used.

Operator response
If the default messages are acceptable, no action is necessary. Otherwise, contact the system programmer to correct the indicated error.

System programmer response
If you want to use the message catalog, correct the indicated error. If the default messages are acceptable, no action is necessary.

User response
Not applicable.

Problem determination
Not applicable.
Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctl.c

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1650I Unable to open message catalog nssctlmsg.cat: errno 113 (EDC5113I Bad file descriptor.)
errnojr 0xC90F0003 - Default messages will be used

EZD1651I Unsupported option opt

Explanation
Parsing for the nssctl command detected an unsupported option.
In the message text:

opt
  The option that is not supported.

System action
The nssctl command processing ends.

Operator response
See the information about the nssctl command in z/OS Communications Server: IP System Administrator's Commands or issue the man nssctl command in a z/OS UNIX shell to obtain information about the nssctl command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command
Module
nssctl.c

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>Message Code</th>
<th>Message Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1651I</td>
<td>Unsupported option -e</td>
</tr>
<tr>
<td>EZD1652I</td>
<td>Option opt is missing a required value</td>
</tr>
</tbody>
</table>

Explanation
The specified nssctl command option requires a value, but no value was specified on the command.

In the message text:

opt
The specified option.

System action
The nssctl command processing ends.

Operator response
Correct and reissue the command. See the information about the nssctl command in z/OS Communications Server: IP System Administrator's Commands or issue the man nssctl command in a z/OS UNIX shell to obtain information about the nssctl command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctl.c
Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1652I Option -D is missing a required value

EZD1653I Option opt value length exceeds limit of limit characters

Explanation
The nssctl command option value that was specified exceeds the character limit.
In the message text:

*opt*  
The option that was specified.

*limit*  
The maximum number of characters allowed for the option value.

System action
The nssctl command processing ends.

Operator response
Correct and reissue the command. See the information about the nssctl command in z/OS Communications Server: IP System Administrator’s Commands or issue the man nssctl command in a z/OS UNIX shell to obtain information about the nssctl command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctl.c

Routing code
Not applicable.
Descriptor code
Not applicable.

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>EZD1653I</th>
<th>Option -c value length exceeds limit of 24 characters</th>
</tr>
</thead>
</table>

EZD1654I  Option opt does not support value val

Explanation
The specified value is not supported for this nssctl command option.

In the message text:

opt
   The option that was specified.

val
   The unsupported value.

System action
The nssctl command processing ends.

Operator response
Correct and reissue the command. See the information about the nssctl command in z/OS Communications Server: IP System Administrator's Commands or issue the man nssctl command in a z/OS UNIX shell to obtain information about the nssctl command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctl.c

Routing code
Not applicable.

Descriptor code
Not applicable.
EZD1655I  Primary option not specified

Explanation
The nssctl command requires a primary option but none was provided.

System action
The nssctl command processing ends.

Operator response
Correct and reissue the command. See the information about the nssctl command in z/OS Communications Server: IP System Administrator's Commands or issue the man nssctl command in a z/OS UNIX shell to obtain information about the nssctl command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctl.c

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
Not applicable.
**Explanation**
The `nssct1` command requires the user to have authorization from the System Authorization Facility (SAF). The requester does not have authority to perform the requested action.

**System action**
The `nssct1` command processing ends.

**Operator response**
Contact the system programmer to obtain the appropriate authority to use the `nssct1` command.

**System programmer response**
Give the user the appropriate authority to use the `nssct1` command. See the information about the `nssct1` command in z/OS Communications Server: IP System Administrator's Commands for more information about the SAF permission required by the `nssct1` command.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX nssct1 command

**Module**
nssct1.c

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**
Not applicable.

---

**EZD1657I Memory could not be obtained to complete the request**

**Explanation**
There is not enough memory to satisfy the `nssct1` command request.

**System action**
The `nssct1` command processing ends.
Operator response
The error might be transient; reissue the request. If the error persists, contact the system programmer

System programmer response
Trace or log entries might provide more information about the error. Ensure that there is enough memory available on the system. See the information about diagnosing storage abends and storage growth in z/OS Communications Server: IP Diagnosis Guide for more information about storage problems.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctl.c

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
Not applicable.

EZD1658I sigaction() failed for signal : errno errno (description)

Explanation
An error occurred during the establishment of a signal handler in support of the nssctl command request.

In the message text:

signal
The signal that the nssctl command attempted to register with the signal handler.

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.

description
Describes the meaning of the errno value.

System action
The nssctl command processing ends.
Operator response
Contact the system programmer.

System programmer response
Correct the error indicated by errno and description, and reissue the command.

User response
Not applicable.

nssctl

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctl.c

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1658I sigaction() failed for SIGABND : errno 121 (EDC5121I Invalid argument.)

EZD1659I The nssctl command is not APF authorized

Explanation
The nssctl command is an APF-authorized application, but the APF bit is not set.

System action
The nssctl command processing ends.

Operator response
Contact the system programmer.

System programmer response
Ensure that the nssctl command is installed correctly. Issue the extattr command to ensure that the APF-authorized attribute is set to ON. See the z/OS UNIX System Services Command Reference for information about the extattr command syntax and options.
**EZD1660I** Extraneous parameter *parm*

**Explanation**
An extraneous parameter was specified on an `nssctl` command.

In the message text:

`parm`  
The extraneous parameter.

**System action**
The `nssctl` command processing ends.

**Operator response**
Remove the extraneous parameter and issue the `nssctl` command again. See the information about the `nssctl` command in *z/OS Communications Server: IP System Administrator’s Commands* or issue the `man nssctl` command in a z/OS UNIX shell to obtain information about the `nssctl` command syntax and options.

**System programmer response**
None

**User response**
Not applicable.
**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX nssctl command

**Module**
nssctl.c

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**

```
EZD1660I Extraneous parameter abc
```

**EZD1663I**  Signal *signum* received - nssctl command processing ended

**Explanation**
A system signal was received, which forced the nssctl command to end. See z/OS UNIX System Services Command Reference for more information about signals.

In the message text:

*signum*
  The signal that was received.

**System action**
The nssctl command processing ended.

**Operator response**
The condition might be temporary; try the command again and specify the debug (-Z) option on the command specification. If the failure persists, contact the system programmer.

**System programmer response**
Contact IBM software support services and provide a syslog that includes the time of the command failure. If available, provide a CTRACE for component SYSTCPIK.

**User response**
Not applicable.

**Problem determination**
Not applicable.
Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctl.c

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1663I Signal SIGTERM received - nssctl command processing ended

EZD1664I Connect error for server daemon connection : function errno errno (description) errnojr errnojr

Explanation
This message is displayed by the z/OS UNIX nssctl command when an error occurred while connecting to the server daemon.

In the message text:

server
The network security services (NSS) daemon to which the nssctl command is trying to connect.

function
The name of the C/C++ runtime library function that detected the error.

erro
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

description
Describes the meaning of the errno value.

errnojr
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The nssctl command processing ends.

Operator response
Contact the system programmer.

System programmer response
Verify that the daemon is running by issuing the MODIFY proclname, DISPLAY command, where proclname is the member name of the cataloged procedure used to start the daemon. Use the information from the function, errno, and errnojr values to fix the problem.
User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctlNMI_Transaction.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
The following message is displayed if the nssctl command tries to connect to a daemon that is not running.

EZD1664I Connect error for NSS daemon connection : connect errno 1128 (EDC8128I Connection refused.)
errnoj 0x120D0253

EZD1665I Send error on server daemon connection : function errno errno (description) errnoj errnoj

Explanation
This message is displayed by the z/OS UNIX nssctl command when an error occurs during a write operation while connecting to the server daemon.

In the message text:

server
The network security services (NSS) daemon to which the nssctl command is trying to connect.

function
The name of the C/C++ runtime library function that detected the error.

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errno) information in z/OS UNIX System Services Messages and Codes.

description
Describes the meaning of the errno value.

ernoj
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnoj) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The nssctl command processing ends.
Operator response
Contact the system programmer.

System programmer response
Verify that the daemon is running by issuing the MODIFY procname, DISPLAY command, where procname is the member name of the cataloged procedure used to start the daemon. Use the information from the function, errno, and errnojr values to fix the problem.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctlNMI_Transaction.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
The following message is displayed if the NSS daemon is stopped while an nssct1 command request is in progress.

EZD1665I Send error on NSS daemon connection: write errno 1124 (EDC8124I Socket not connected.) errnojr 0x120D025C

EZD1666I Receive error on server daemon connection: function errno errno (description) errnojr errnojr

Explanation
This message is displayed by the z/OS UNIX nssct1 command when an error occurs during a read operation while connecting from the server daemon.

In the message text:

server
The network security services (NSS) daemon to which the nssct1 command is trying to connect.

function
The name of the C/C++ runtime library function that detected the error.

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.

1098 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**description**
Describes the meaning of the `errno` value.

**errnojr**
The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (`errnojr`) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
The `nssctl` command processing ends.

**Operator response**
Contact the system programmer

**System programmer response**
Verify that the daemon is running by issuing the MODIFY `procname`, DISPLAY command, where `procname` is the member name of the cataloged procedure used to start the daemon. Use the information from the `function`, `errno`, and `errnojr` values to fix the problem.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX nssctl command

**Module**
nssctlNMI_Transaction.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**
The following message is displayed if the NSS daemon or IKE daemon is stopped while a `nssctl` command request is in progress.

```
EZD1666I Receive error on NSS daemon connection : read errno 1124 (EDC8124I Socket not connected.)
errnojr 0x120D025C
EZD1667I Data received over the server daemon connection is not valid : return code returncode
```
Explanation
This message is displayed by the z/OS UNIX nssctl command when data received from the daemon is not in the expected format.

In the message text:

**server**
The daemon whose connection received the illegal data.

**returncode**
Possible values are:

1. The message header identifier is not valid.
2. The message header version number is not valid.
3. The message type is not valid.
4. The message header size is not valid.
5. The message size is not valid.
6. The message contains a reserve area that is not set to zeros.
7. The message contains a record length that is not valid.
8. The message contains a record count that is not valid.
9. The message contains a section with a length that is not valid.
10. The message contains a section with a count field that is not valid.
14. The response message contains a correlation ID that does not match the correlation ID in the request message.
15. The response message contains a message type that does not correspond to the message type in the request message.

System action
The nssctl command processing ends.

Operator response
Contact the system programmer.

System programmer response
Issue the nssctl command again with the -Z option to determine the sequence of events leading up to the error. Contact the IBM Software Support Center.

User response
Not applicable.
Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctlNMI_Transaction.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
The following message is displayed if the response message received from the daemon does not correspond to the request message sent to the daemon.

EZD1667I Data received over the NSS daemon connection is not valid : return code 14

EZD1668I Message received from server daemon with return code returncode ( description ) reason code reasoncode

Explanation
The server daemon returned a response message with a nonzero return code.

In the message text:

server
The NSS daemon from which the nssctl command received the message.

returncode
The return code contained in the response message. See the information about network manager return and reason codes in z/OS Communications Server: IP Programmer's Guide and Reference and the diagnosing network security services (NSS) server problems information in z/OS Communications Server: IP Diagnosis Guide to determine the appropriate response.

description
Describes the meaning of the returncode value.

reasoncode
The reason code contained in the response message. See the information about network manager return and reason codes in z/OS Communications Server: IP Programmer's Guide and Reference and the information about diagnosing network security services (NSS) server problems in z/OS Communications Server: IP Diagnosis Guide to determine the appropriate response. For reason codes that have a mnemonic starting with NSS, look first in the z/OS Communications Server: IP Programmer's Guide and Reference. For reason codes that have a mnemonic starting with the letters NM, look first in the z/OS Communications Server: IP Diagnosis Guide.
System action
The `nssctl` command processing ends.

Operator response
Contact the system programmer

System programmer response
Correct the problem using the network manager return and reason codes in z/OS Communications Server: IP Programmer's Guide and Reference or using the information about diagnosing network security services (NSS) server problems in z/OS Communications Server: IP Diagnosis Guide.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctlNMI_Transaction.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
```
EZD1668I Message received from NSS daemon with return code 121 ( EDC5121I  Invalid argument. )
reason code NMsRsnInvalidAPIVersion
```

EZD1669I   Internal nssctl command error : return code `returncode`

Explanation
This message is displayed by the z/OS UNIX `nssctl` command when an internal error is detected. In the message text:

`returncode`
- The return code from the `nssctl` command.

System action
The `nssctl` command processing ends.
**Operator response**
Contact the system programmer.

**System programmer response**
Issue the `nssctl` command again with the -Z option to determine the sequence of events leading up to the error. Contact the IBM Software Support Center.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX nssctl command

**Module**
nssctlNMI_Transaction.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**

```
EZD1669I Internal nssctl command error : return code 1
```

**EZD1670I Client clientname is not available**

**Explanation**
The Network Security Services (NSS) client specified with the `nssctl` command -c option cannot be found.

In the message text:

`clientname`
The name of the NSS client that was specified with the -c option of the z/OS UNIX `nssctl` command.

**System action**
The `nssctl` command processing ends.

**Operator response**
Determine whether the NSS client is connected to the NSS server by issuing the z/OS UNIX `nssctl -d` command on the system where the NSS daemon is running.
System programmer response

- For an NSS IPSec client, determine whether there is an NssStackConfig statement that defines the client in the IKE daemon configuration file on the system where the client is running.

  See the information about the IKE daemon in z/OS Communications Server: IP Configuration Reference and the information about IP security in z/OS Communications Server: IP Configuration Guide for information about configuring the NssStackConfig statement.

- For an NSS XMLAppliance client, see the XML appliance documentation for NSS server configuration.

User response

Not applicable.

Problem determination

None

Source

z/OS Communications Server: z/OS UNIX nssctl command

Module

nssctlNMI_Transaction.cpp

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

Example

Ezd1670I Client MVS134_TPCS3 is not available

Ezd1671I Incorrect opt option value value is ignored

Explanation

An incorrect option value was specified and ignored.

In the message text:

- **opt**
  - The `nssctl` command option that was specified

- **value**
  - The value that was specified for the `nssctl` command option.

System action

The `nssctl` command continues, using any specified valid values or any default values.
**Operator response**

Specify an option value that is in the accepted value range and issue the `nssctl` command again. See the information about the `nssctl` command in z/OS Communications Server: IP System Administrator's Commands or issue the `man nssctl` command in a z/OS UNIX shell to obtain information about the `nssctl` command syntax and options.

**System programmer response**

None

**User response**

Not applicable.

**Problem determination**

None

**Source**

z/OS Communications Server: z/OS UNIX nssctl command

**Module**

nssctl.c

**Routing code**

Not applicable.

**Descriptor code**

Not applicable.

**Automation**

Not applicable.

**Example**

```
EZD1671I Incorrect -Z option value 44 is ignored
```

**EZD1672I**  Primary options `opt1` and `opt2` cannot both be specified

**Explanation**

The `nssctl` command can specify only one primary option at a time.

In the message text:

`opt1` and `opt2`

The primary options that were specified.

**System action**

The `nssctl` command processing ends.
Operator response

Issue the `nssctl` command again with only one primary option. See the information about the `nssctl` command in z/OS Communications Server: IP System Administrator's Commands or issue the `man nssctl` command in a z/OS UNIX shell to obtain information about the `nssctl` command syntax and options.

System programmer response

None

User response

Not applicable.

Problem determination

None

Source

z/OS Communications Server: z/OS UNIX nssctl command

Module

nssctl.c

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

Example

```
EZD1672I Primary options -d and -? cannot both be specified
```

```
EZD1673I Filter options opt1 and opt2 cannot both be specified
```

Explanation

The `nssctl` command can specify only one filter option at a time.

In the message text:

`opt1` and `opt2`

- The filter options that were specified.

System action

The `nssctl` command processing ends.

Operator response

Issue the `nssctl` command again with only one filter option. See the information about the `nssctl` command in z/OS Communications Server: IP System Administrator's Commands or issue the `man nssctl` command in a z/OS UNIX shell to obtain information about the `nssctl` command syntax and options.
System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server: z/OS UNIX nssctl command

Module
nssctl.c

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

EZD1673I Filter options -c and -D cannot both be specified

EZD1721I Packet denied by defensive filter: timestamp filter rule= rulename ext= instance sipaddr= sipaddr dipaddr= dipaddr proto= proto tag1 tag2 tag3 Interface= ifcaddr (dir) secclass= secclass dest= dest len= len ifcname= ifcname fragment= frag

Explanation
An IP packet matched the indicated defensive filter rule and was denied. For this message to be written, the matching defensive filter must have logging enabled.

In the message text:

*timestamp*
  The stack timestamp that indicates the time at which the IP packet was processed by the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

*rulename*
  The defensive filter rule name as specified on the -N option when the defensive filter was added with the z/OS UNIX *ipsec* command.

*instance*
  The rule name extension that indicates which instance of the rule name was matched.

*sipaddr*
  The source IP address of the packet.

*dipaddr*
  The destination IP address of the packet.
proto
The protocol from the packet. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
- AH(51)
- ICMPv6(58)
- OSPF(89)
- IPIP(94)
- MIPv6(135)
- Unknown
- The protocol number

tag1
The tag1 value varies depending on the proto value.

- If the proto value is ICMP or ICMPv6, the tag1 value is type= followed by the ICMP or ICMPv6 type, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the proto value is TCP or UDP, the tag1 value is sport= followed by the source port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
- If the proto value is OSPF, the tag1 value is type= followed by the type, or followed by the value Unknown if the OSPF header is not present in the packet as the result of fragmentation.
- If the proto value is MIPv6, the tag1 value is type= followed by the type, or followed by the value Unknown if the MIPv6 header is not present in the packet as the result of fragmentation.
- If the proto value is any value not previously mentioned, the tag1 value is -= which indicates that the data is not applicable.

tag2
The tag2 value varies depending on the proto value.

- If the proto value is ICMP or ICMPv6, the tag2 value is code= followed by the ICMP or ICMPv6 code, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the proto value is TCP or UDP, the tag2 value is dport= followed by the destination port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
- If the proto value is any value not previously mentioned, the tag2 value is -= which indicates that the data is not applicable.

tag3
The tag3 value varies depending on the proto value and direction.

- If the proto value is TCP or UDP, the direction is inbound, and the port has been translated by the CommServer NAT Traversal function, the tag3 value is origport= followed by the original source port.
- If the proto value is TCP or UDP, the direction is outbound, and the port has been translated by the CommServer NAT Traversal function, the tag3 value is origport= followed by the original destination port.
- If the proto value is any value not previously mentioned, the tag3 value is -= which indicates that the data is not applicable.

ifcaddr
The interface address over which the packet was received or sent.
dir
   Possible values are:
   I
      The packet is inbound.
   O
      The packet is outbound.
secclass
   The security class assigned to the interface. The security class is a numeric value in the range of 1-255.
dest
   Possible values are:
   local
      The destination is a local destination.
   routed
      The packet is being routed.
len
   The packet length.
ifcname
   The interface name.
frag
   Possible values are:
   Y
      The packet is a fragment.
   N
      The packet is not a fragment.
   routed
      The packet is not a fragment.

System action
TCP/IP processing continues.

Operator response
No action needed.

System programmer response
No action needed.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: TRMD

Module
EZATRZOS
Routing code
Not applicable for syslog message.

Descriptor code
Not applicable for syslog message.

Automation
Not applicable.

Example
Block_192.30.30.0/24
  ext= 1 sipaddr= 192.30.30.1 dipaddr= 192.1.1.1 proto= tcp(6) sport= 65000 dport= 21 -=
  Interface= 192.1.1.1 (I) secclass= 255 dest= local len= 88 ifcname= LINK1 fragment= N

EZD1722I Packet would have been denied by defensive filter: timestamp filter rule= rulename ext=
  instance sipaddr= sipaddr dipaddr= dipaddr proto= proto tag1 tag2 tag3 Interface= ifcaddr
  (dir ) secclass= secclass dest= dest len= len ifcname= ifcname fragment= frag

Explanation
An IP packet matched the indicated defensive filter rule and the defensive filter mode is simulate. You can set
the defensive filter mode to simulate for a single defensive filter or you can set the mode to simulate for the
stack, which overrides the mode on the individual defensive filter rules.

Use the simulate mode to monitor the impact of enabling defensive filters without discarding traffic. When an IP
packet matches a defensive filter and the mode is simulate, this message is logged, which indicates that the
packet would have been discarded if the mode had been block. The packet is not discarded and IP filtering
continues.

In the message text:

**timestamp**
The stack timestamp that indicates the time at which the IP packet was processed by the stack. This time is
retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This
timestamp might be different than the syslogd message timestamp.

**rulename**
The defensive filter rule name as specified on the -N option when the defensive filter was added with the
z/OS UNIX **ipsec** command.

**instance**
The rule name extension that indicates which instance of the rule name was matched.

**sipaddr**
The source IP address of the packet.

**dipaddr**
The destination IP address of the packet.

**proto**
The protocol from the packet. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
• AH(51)
• ICMPv6(58)
• OSPF(89)
• IPIP(94)
• MIPv6(135)
• Unknown

The protocol number

**tag1**
The *tag1* value varies depending on the *proto* value.

- If the *proto* value is ICMP or ICMPv6, the *tag1* value is `type=` followed by the ICMP or ICMPv6 type, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the *proto* value is TCP or UDP, the *tag1* value is `sport=` followed by the source port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
- If the *proto* value is OSPF, the *tag1* value is `type=` followed by the type, or followed by the value Unknown if the OSPF header is not present in the packet as the result of fragmentation.
- If the *proto* value is MIPv6, the *tag1* value is `type=` followed by the type, or followed by the value Unknown if the MIPv6 header is not present in the packet as the result of fragmentation.
- If the *proto* value is any value not previously mentioned, the *tag1* value is `-=` which indicates that the data is not applicable.

**tag2**
The *tag2* value varies depending on the *proto* value.

- If the *proto* value is ICMP or ICMPv6, the *tag2* value is `code=` followed by the ICMP or ICMPv6 code, or followed by the value Unknown if the ICMP header is not present in the packet as the result of fragmentation.
- If the *proto* value is TCP or UDP, the *tag2* value is `dport=` followed by the destination port, or followed by the value Unknown if the TCP or UDP header is not present in the packet as the result of fragmentation.
- If the *proto* value is any value not previously mentioned, the *tag2* value is `-=` which indicates that the data is not applicable.

**tag3**
The *tag3* value varies depending on the *proto* value and direction.

- If the *proto* value is TCP or UDP, the direction is inbound, and the port has been translated by the CommServer NAT Traversal function, the *tag3* value is `origport=` followed by the original source port.
- If the *proto* value is TCP or UDP, the direction is outbound, and the port has been translated by the CommServer NAT Traversal function, the *tag3* value is `origport=` followed by the original destination port.
- If the *proto* value is any value not previously mentioned, the *tag3* value is `-=` which indicates that the data is not applicable.

**ifcaddr**
The interface address over which the packet was received or sent.

**dir**
Possible values are:

I  
The packet is inbound.

O  
The packet is outbound.

**secclass**
The security class assigned to the interface. The security class is a numeric value in the range of 1-255.
**dest**
Possible values are:

**local**
The destination is a local destination.

**routed**
The packet is being routed.

**len**
The packet length.

**ifc**
The interface name.

**frag**
Possible values are:

**Y**
The packet is a fragment.

**N**
The packet is not a fragment.

**routed**
The packet is not a fragment.

**System action**
TCP/IP processing continues.

**Operator response**
No action needed.

**System programmer response**
No action needed.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: TRMD

**Module**
EZATRZOS

**Routing code**
Not applicable for syslog message.

**Descriptor code**
Not applicable for syslog message.
Automation
Not applicable.

Example

EZD1722I Packet would have been denied by defensive filter: 07/11/2007 23:40:08.78 filter rule=
  Block_192.30.30.0/24 ext= 1 sipaddr= 192.30.30.1 dipaddr= 192.1.1.1 proto= tcp(6)
  sport= 65000 dport= 21 -= Interface= 192.1.1.1 (I) secclass= 255 dest= local len= 88
  ifcname= LINK1 fragment= N

EZD1723I Defensive filter added: date time filter rule= rulename ext= instance sipaddr= sipaddr / sip_prefix_length dipaddr= dipaddr / dip_prefix_length proto= proto tag1 tag2 fragments_only= fragments_only dir= dir routing= routing mode= mode log= log lifetime= lifetime userid= userid
global= global_setting loglimit= loglimit

Explanation

A defensive filter is added to the TCP/IP stack.

In the message text:

**date**
The date on which the defensive filter was added to the stack. This date is retrieved from the system time-of-
day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than
the syslogd message timestamp.

**time**
The time at which the defensive filter was added to the stack. This time is retrieved from the system time-of-
day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than
the syslogd message timestamp.

**rulename**
The defensive filter rule name as specified on the -N option when the defensive filter was added with the
z/OS UNIX **ipsec** command.

**instance**
The rule name extension.

**sipaddr / sip_prefix_length**
The source IP address specification for the defensive filter rule. The value 0.0.0.0/0 indicates that the
defensive filter rule applies to all source IPv4 addresses. The value ::/0 indicates that the defensive filter rule
applies to all source IPv6 addresses.

**dipaddr / dip_prefix_length**
The destination IP address specification for the defensive filter rule. The value 0.0.0.0/0 indicates that the
defensive filter rule applies to all destination IPv4 addresses. The value ::/0 indicates that the defensive filter
rule applies to all destination IPv6 addresses.

**proto**
The protocol specification for the defensive filter rule. Possible values are:

- ICMP(1)
- IGMP(2)
- IP(4)
- TCP(6)
- UDP(17)
- ESP(50)
- AH(51)
- ICMPv6(58)
- OSPF(89)
- IPIP(94)
• MIPv6(135)
• The protocol number
• ALL

tag1
  The tag1 value varies depending on the proto value.
  • If the proto value is ICMP or ICMPv6, the tag1 value is type= followed by the ICMP or ICMPv6 type, or followed by the value all.
  • If the proto value is TCP or UDP, the tag1 value is sport= followed by the source port range. For example, sport= 1024-65535. For a defensive filter that applies to all source ports the tag1 value is sport= 1-65535.
  • If the proto value is any value not previously mentioned, the tag1 value is -= which indicates that the data is not applicable.

tag2
  The tag2 value varies depending on the protocol.
  • If the proto value is ICMP or ICMPv6, the tag2 value is code= followed by the ICMP or ICMPv6 code, or followed by the value all.
  • If the proto value is TCP or UDP, the tag2 value is dport= followed by the destination port range. For example, dport= 21-21. For a defensive filter that applies to all destination ports, the tag2 value is dport= 1-65535.
  • If the proto value is any value not previously mentioned, the tag2 value is -= which indicates that the data is not applicable.

fragments_only
  Possible values are:
  yes
    The defensive filter rule applies only to fragments.
  no
    The defensive filter rule does not apply only to fragments.

dir
  The direction specified for the defensive filter rule. Possible values are inbound and outbound.

routing
  The routing specified for the defensive filter rule. Possible values are local, routed, and either.

mode
  The defensive filtering mode specified for the defensive filter rule. Possible values are block and simulate.

log
  The log specified for the defensive filter rule. Possible values are yes and no. If the mode value is Simulate, the log value is not applicable and logging is always performed.

lifetime
  The lifetime of the defensive filter rule in minutes.

userid
  The user ID of the user who added the defensive filter rule.

global_setting
  Possible values are:
  yes
    The defensive filter rule was created as a global filter rule.
  no
    The defensive filter rule was created as a stack-specific filter rule.

loglimit
  The limit on the number of filter-match messages generated for this filter in a 5-minute interval. A value of 0 indicates that there is no limit.
System action
TCP/IP processing continues.

Operator response
No action needed.

System programmer response
No action needed.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: TRMD

Module
EZATRZOS

Routing code
Not applicable for syslog message.

Descriptor code
Not applicable for syslog message.

Automation
Not applicable.

Example
EZD1723I Defensive filter added: 07/11/2012 23:40:08.78 filter rule= Block_192.30.30.0/24 ext= 1 sipaddr= 192.30.30.0 / 24 dipaddr= 0.0.0.0 / 0 proto= tcp(6) sport= 1024 - 65535 dport= 21 fragmentsonly= no dir= inbound routing= local mode= block log= yes lifetime= 30 userid= USER1
global= no loglimit= 100

EZD1724I Defensive filter deleted: timestamp filter rule= rulename ext= instance reason= reason userid= userid

Explanation
A defensive filter is deleted from the TCP/IP stack.

In the message text:

timestamp
The stack timestamp that indicates the time at which the defensive filter was deleted from the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.
**rulename**
The defensive filter rule name as specified on the -N option when the defensive filter was added with the z/OS UNIX **ipsec** command.

**instance**
The rule name extension.

**reason**
The reason that the defensive filter was deleted from the TCP/IP stack. Possible **reason** values are:
- **expire**
  The defensive filter lifetime expired and the filter was deleted from the stack.
- **delete_specific**
  The defensive filter was deleted because a z/OS UNIX **ipsec** command was issued to delete this filter by name.
- **delete_all**
  The defensive filter was deleted because a z/OS UNIX **ipsec** command was issued to delete all the defensive filters on this stack.
- **defensive_mode_inactive**
  The defensive filter was deleted because the user changed the defensive filter mode to inactive. The defensive filter mode can be set to inactive by editing the Defense Manager daemon (DMD) configuration file or by issuing the MODIFY procname,FORCE_INACTIVE command.

**userid**
The user ID of the user who deleted the defensive filter. If the **reason** value is expire or defensive_mode_inactive, the **userid** value is N/A.

**System action**
TCP/IP processing continues.

**Operator response**
No action needed.

**System programmer response**
No action needed.

**User response**
Not applicable.

**Problem determination**
Not applicable

**Source**
z/OS Communications Server TCP/IP: TRMD

**Module**
EZATRZOS

**Routing code**
Not applicable for syslog message.
**Descriptor code**
Not applicable for syslog message.

**Automation**
Not applicable.

**Example**

```
EZD1724I Defensive filter deleted: 07/11/2007 23:40:08.78 filter rule= Block_192.30.30.0/24
ext= 1 reason= delete_specific userid= USER2

EZD1725I Defensive filter updated: date time filter rule= rulename ext= instance mode= mode log= log
lifetime= lifetime userid= userid loglimit= loglimit
```

**Explanation**
One or more values have been updated for a defensive filter in the TCP/IP stack. These updates were made using the z/OS UNIX `ipsec` command.

In the message text:

- `date`
  The date on which the defensive filter was updated in the stack. This date is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

- `time`
  The time at which the defensive filter was updated in the stack. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

- `rulename`
  The defensive filter rule name as specified on the `-N` option when the defensive filter was added with the z/OS UNIX `ipsec` command.

- `instance`
  The rule name extension.

- `mode`
  The defensive filtering mode specification for the defensive filter. If the defensive filtering mode was updated with the z/OS UNIX `ipsec` command, the `mode` value is `block` or `simulate`. If the defensive filtering mode was not updated, the `mode` value is `N/A`.

- `log`
  The log specification for the defensive filter. If the log setting was updated with the z/OS UNIX `ipsec` command, the `log` value is `yes` or `no`. If the log setting was not updated, the `log` value is `N/A`.

- `lifetime`
  The lifetime of the defensive filter in minutes. If the lifetime was updated with the z/OS UNIX `ipsec` command, the `lifetime` value is the new lifetime value. If the lifetime was not updated, the value is `N/A`.

- `userid`
  The user ID of the user who updated the defensive filter.

- `loglimit`
  The limit on the number of filter-match messages generated for this filter in a 5-minute interval. A value of 0 indicates that there is no limit. If the log limit was not updated, the value is `N/A`.

**System action**
TCP/IP processing continues.
Operator response
No action needed.

System programmer response
No action needed.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: TRMD

Module
EZATRZOS

Routing code
Not applicable for syslog message.

Descriptor code
Not applicable for syslog message.

Automation
Not applicable.

Example
EZD1725I Defensive filter updated: 07/11/2012 23:40:08.78 filter rule= Block_192.30.30.0/24 ext= 1
mode=
simulate log= N/A lifetime= N/A userid= USER1 loglimit= 100

EZD1726I  SWSA shadow tunnel installation failed timestamp vpaction= vpnaction tunnelID= tunID
AHSPI=AHIndex ESPSPI=ESPindex reason= rsn reason code= rsncode

Explanation
The installation of a Sysplex-Wide Security Associations (SWSA) shadow tunnel for a distributed DVIPA on a
target stack failed.

The SWSA function enables a distributing TCP/IP stack to negotiate IPSec tunnels for distributed DVIPAs. The
IPSec tunnels are installed on target stacks for the distributed DVIPA as shadow tunnels. SWSA requires that the
IP filter policy that applies to the DVIPA be consistent between the target stack and the distributing stack. One
reason the tunnel installation might fail is if the IP filter policies are not consistent, therefore the shadow tunnel
cannot be installed because the target stack does not have IP filter rules that correspond to that tunnel.

In the message text:

timestamp
Indicates when the installation failure occurred. This time is retrieved from the system time-of-day clock,
which usually reflects coordinated universal time (UTC). This time stamp might be different than the syslogd
message time stamp.
vpnaction
The vpnaction name.

- If configured with the IBM Configuration Assistant for z/OS Communications Server, the vpnaction name corresponds to the name of the security level in the GUI. The vpnaction name also contains a suffix appended to the security level name to guarantee uniqueness.
- If configured in the Policy Agent configuration file, the vpnaction name is the name specified on the IpDynVpnAction statement.

tunID
The tunnel ID.

AHindexis
The AH security parameter index.

ESPindexis
The ESP security parameter index.

rsn
Indicates the specific reason the installation failed.

<table>
<thead>
<tr>
<th>reason</th>
<th>Explanation</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Error encountered while you try to install the shadow tunnel.</td>
<td>See the reason codes below for additional information.</td>
</tr>
<tr>
<td>2</td>
<td>Error encountered while you try to install the dynamic filter associated with the shadow tunnel.</td>
<td>See the reason codes below for additional information.</td>
</tr>
<tr>
<td>3</td>
<td>Target stack is FIPS140 enabled, and the tunnel was not negotiated in FIPS140 compliant mode.</td>
<td>A target stack that is enabled for FIPS140 will not accept a tunnel from a distributing stack that is not enabled for FIPS140.</td>
</tr>
</tbody>
</table>

rsncode
Provides additional information about the installation failure.

<table>
<thead>
<tr>
<th>reason code</th>
<th>Explanation</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No additional information provided.</td>
<td>This value is only applicable when reason has a value of 3.</td>
</tr>
<tr>
<td>7</td>
<td>The dynamic filter did not match an anchor filter.</td>
<td>Ensure that the policy on the target stack is consistent with the policy on the distributing stack.</td>
</tr>
<tr>
<td>8</td>
<td>Default policy is in use.</td>
<td>The shadow tunnel cannot be installed if the policy from the policy agent is not currently in use.</td>
</tr>
<tr>
<td>24</td>
<td>ICSF failure occurred.</td>
<td>This message will be preceded by message EZD1730I that indicates the return and reason Codes from ICSF. See the ICSF and TSS Return and Reason Codes in z/OS Cryptographic Services ICSF Application Programmer’s Guide for the specific actions to be taken.</td>
</tr>
<tr>
<td>25</td>
<td>ICSF is not active.</td>
<td>Services from ICSF are required to install this shadow tunnel. ICSF must be started.</td>
</tr>
<tr>
<td>114</td>
<td>Tunnel could not be added to internal structures because of duplicates.</td>
<td>Contact the IBM Software Support Center.</td>
</tr>
<tr>
<td>reason code</td>
<td>Explanation</td>
<td>Comments</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>121</td>
<td>The authentication algorithm provided is not supported.</td>
<td>Ensure that the policy on the target stack is consistent with the policy on the distributing stack. Some algorithms are not supported because of export restrictions. Ensure that the algorithm that is being used is supported.</td>
</tr>
<tr>
<td>132</td>
<td>Storage shortage</td>
<td>Storage to complete the request is not currently available. Determine the cause of the storage failure.</td>
</tr>
<tr>
<td>134</td>
<td>The encryption algorithm provided is not supported.</td>
<td>Ensure that the policy on the target stack is consistent with the policy on the distributing stack. Some encryption algorithms are not supported because of export restrictions. Ensure that the algorithm that is being used is supported.</td>
</tr>
<tr>
<td>1008</td>
<td>The dynamic filter being added conflicted with an existing dynamic filter.</td>
<td>Ensure that the policy on the target stack is consistent with the policy on the distributing stack.</td>
</tr>
</tbody>
</table>

See the information about Sysplex-wide Security Associations in z/OS Communications Server: IP Configuration Guide.

System action
The tunnel installation fails; TCP/IP processing continues.

Operator response
None.

System programmer response
Ensure that the IP filter policy on the distributing stack for all traffic pertaining to the distributed DVIPA is correctly mirrored on the target stack. Also, take any additional action dictated by the reason code.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IPSec

Module
ezatrzos.c

Routing code
Not applicable.

Descriptor code
Not applicable.
Example

EZD1726I  SWSA shadow tunnel installation failed: 04/30/2009 19:47:27.99 vpnaction= IPSec__Gold
tunnelID= Y4 AHSPID= 0 ESPSPI= 3517985610 reason= 3 reason code= 0

EZD1727I  FIPS140 support is enabled for IPSec

Explanation

The Federal Information Processing Standard 140 (FIPS 140) function is enabled for IPSec in the TCP/IP stack. All cryptographic operations are performed by cryptographic modules that are designed to follow the Level 1 security requirements of FIPS publication 140-2.

System action

TCP/IP processing continues.

Operator response

No action needed.

System programmer response

No action needed.

User response

Not applicable.

Problem determination

Not applicable

Source

z/OS Communications Server TCP/IP: TRMD

Module

ezatrzos.c

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

Example

EZD1727I  FIPS140 support is enabled for IPSec
EZD1728I  FIPS140 support is not enabled for IPSec

Explanation
The Federal Information Processing Standard 140 (FIPS 140) function is not enabled for IPSec in the TCP/IP stack. Cryptographic operations might be performed by cryptographic modules that are not designed to follow the Level 1 security requirements of FIPS publication 140-2.

System action
TCP/IP processing continues.

Operator response
No action needed.

System programmer response
If FIPS 140 support is required for IPSec, then configure FIPS140 Yes on the IpFilterPolicy statement; otherwise, no action is needed.

User response
Not applicable.

Problem determination
Not applicable

Source
z/OS Communications Server TCP/IP: TRMD

Module
ezatrzos.c

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1728I FIPS140 support is not enabled for IPSec

EZD1730I  ICSF Service CSFPTRC failed: timestamp ICSF Return Code= return_code ICSF Reason Code= reason_code

Explanation
The Integrated Cryptographic Service Facility (ICSF) service CSFPTRC returned an error. This service is called to create a session object.
In the message text:

**timestamp**
The time at which the failure occurred. This time is retrieved from the system time-of-day clock, which usually reflects coordinated universal time (UTC). This timestamp might be different than the syslogd message timestamp.

**return_code**
The return code value, in hexadecimal format, that was returned by the ICSF service.

**reason_code**
The reason code value, in hexadecimal format, that was returned by the ICSF service.

**System action**
The cryptographic request fails and TCP/IP processing continues.

**Operator response**
None.

**System programmer response**
See the information about the ICSF and TSS Return and Reason Codes in z/OS Cryptographic Services ICSF Application Programmer's Guide for the specific actions to be taken.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: TRMD

**Module**
ezatrzos.c

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**

```
```

```
EZD1731I  The Defense Manager daemon marked the persistent filter data for stack **stackname** as untrusted
```
Explanation
When the Defense Manager daemon (DMD) was restarted, an error was detected in the persistent filter data for
the stack specified by stackname. The persistent filter data file that was named active.stackname was renamed
to untrusted.stackname.timestamp.

In the message text:

**stackname**
The name of the TCP/IP stack for which the persistent filter data is not trusted.

System action
Defense Manager daemon (DMD) processing continues.

Operator response
Contact the system programmer.

System programmer response
The persistent filter data is maintained for each stack in binary files created by the DMD. The files are created in
the directory specified by the DefensiveFilterDirectory keyword on the DmConfig statement in the DMD
configuration file. These binary files are managed by the DMD and must not be modified manually. See the
information about the DefensiveFilterDirectory parameter in the DmConfig statement in z/OS Communications
Server: IP Configuration Reference for information about the persistent filter data files.

This message can be the result of a change to the DefensiveFilterDirectory value, a damaged file system, or
manual editing of the persistent filter data file. Restore the valid persistent data, if available, and restart the
DMD. For example, if the DefensiveFilterDirectory value was changed and the new directory had an outdated
persistent data file for the stack, copy the persistent data from the previous directory to the new directory or
restore the DefensiveFilterDirectory value to its previous value.

If you cannot restore valid persistent data (for example, the file system is damaged), any defensive filters in the
stack remain in effect. However, the DMD has no knowledge of the filters. As a result, you cannot update or
delete the filters by name. Also, if the stack ends, the DMD cannot reinstall the filters in the stack when the stack
is restarted. You can delete all defensive filters from the stack with the **ipsec -F delete -N all** command
if you do not want the filters to remain in effect until their lifetime expiration. New defensive filters can be
successfully added, updated, and deleted.

If you are starting the DMD following the migration and IPL of your system to a new z/OS release, this message
might be the result of a change to the internal structure of the persistent filter data. Defensive filters in the
persistent data cannot be processed and are not installed. New defensive filters can be successfully added,
updated, and deleted. No action is necessary.

If the persistent filter data in the directory specified by DefensiveFilterDirectory appears to be valid, contact IBM
support services. Provide a dump of the DMD and the file named **untrusted.stackname.timestamp**.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1731I The Defense Manager daemon marked the persistent filter data for stack TCPCS as untrusted

EZD1732I AN ERROR OCCURRED WHILE TRYING TO CREATE THE DMD /var/dm/dmd.sock FILE - ERRNO errno description

Explanation
An error occurred when the Defense Manager daemon (DMD) tried to create the /var/dm/dmd.sock file.

In the message text:

errno
The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (errno) information in z/OS UNIX System Services Messages and Codes.

description
Describes the meaning of the errno value.

System action
DMD initial startup processing fails and the DMD ends.

Operator response
Contact the system programmer.

System programmer response
The DMD user ID must have the authority to create, delete, read, and write the /var/dm/dmd.sock file. The most probable cause of the error is that the DMD user ID does not have the authority to create, delete, read, and write files in the /var/dm directory. See the information about configuring the DMD in z/OS Communications Server: IP Configuration Guide. After correcting the error, restart the DMD.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
Not applicable.
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>EZD1732I</th>
<th>AN ERROR OCCURRED WHILE TRYING TO CREATE THE DMD /var/dm/dmd.sock FILE - ERRNO 111 Permission denied.</th>
</tr>
</thead>
</table>

EZD1733I DISPLAY DMD CONFIGURATION:

Explanation
The Defense Manager daemon (DMD) received the MODIFY DMDproc,DISPLAY subcommand. See the information about the Defense Manager daemon MODIFY command in z/OS Communications Server: IP System Administrator's Commands.

System action
The Defense Manager daemon (DMD) continues processing the MODIFY DISPLAY command.

Operator response
None.

System programmer response
None.

User response
No action needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: Defense Manager daemon

Module
DmModifyCommandHandler.cpp

Routing code
2

Descriptor code
5,8,9
Automation
This message is written to both the operator console and syslog.

Example

EZD1733I DISPLAY DMD CONFIGURATION:

EZD1751I exchange_type exchange retransmit timed out from src_ip port src_port to dest_ip port dest_port

Explanation
The Internet Key Exchange (IKE) daemon exhausted the retransmit limit set for an Internet Key Exchange version 2 (IKEv2) exchange retransmission.

Additional diagnostic messages that have the same message instance number are issued to identify the affected Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:

*exchange_type*
  The type of the IKEv2 exchange.

*src_ip*
  The local security endpoint IP specification.

*src_port*
  The source port.

*dest_ip*
  The remote security endpoint IP specification.

*dest_port*
  The destination port.

System action
The IKEv2 exchange fails; IKE daemon processing continues.

Operator response
None.

System programmer response
Check the connectivity to the remote security endpoint. If connectivity exists, then the remote security endpoint is not responding.

User response
None

Problem determination
None

Source
z/OS Communications Server TCP/IP: IKE daemon
Module
IKEv2Exchange.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1751I IKE_SA_INIT exchange retransmit timed out from 9.1.2.3 port 500 to 9.4.5.6 port 500

EZD1752I No applicable policy for IKEv2 negotiation using policy_object_type policy_object_name

Explanation
The Internet Key Exchange (IKE) daemon is attempting to negotiate an Internet Key Exchange version 2 (IKEv2) Security Association (SA), but the policies defined for that SA are not allowed for IKEv2. This situation can occur when the local policy is written for IKEv1, but the IKE peer uses IKEv2 for SA negotiation. For example, an IpDynVpnAction statement must include at least one IpDataOffer specification. For IKEv2, IpDataOffer specifications that include a HowToEncrypt parameter value other than DoNot and a HowToAuth parameter value of AH are omitted at SA negotiation time because IKEv2 does not support this combination. If all of the IpDataOffer specifications are omitted from the IpDynVpnAction statement due to this restriction, the IKE daemon fails IKEv2 SA negotiation and issues this message.

In the message text:

policy_object_type
   The type of the incomplete policy object.

policy_object_name
   The name of the incomplete policy object.

System action
The SA negotiation fails; IKE daemon processing continues.

Operator response
None.

System programmer response
If you did not use the IBM Configuration Assistant for z/OS Communications Server to configure your policy, ensure that the specified policy object has the correct IpDynVpnAction objects for an IKEv2 negotiation. If you did use the IBM Configuration Assistant for z/OS Communications Server to configure your policy, run the Configuration Assistant Health Check to locate policy configuration errors.

User response
None.
Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2DomainOfInterpretation.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1752I No applicable policy for IKEv2 negotiation using IpDynVpnAction AHandESP
EZD1753I IKE version ike_version exchange exchange_type message message_id from remote_ip port remote_port to local_ip port local_port is incorrect because reason

Explanation
The specified message is ignored for the given reason.

In the message text:
ike_version
The version of Internet Key Exchange (IKE) from the message header.

exchange_type
The exchange field value in the IKE message header.

message_id
The message ID in the message header.

remote_ip
The remote security endpoint IP specification.

remote_port
The remote port of the IKE daemon peer.

local_ip
The local security endpoint IP specification.

local_port
The local port of the IKE daemon.

reason
The textual description of why the message is incorrect.

Additional diagnostic messages that have the same message instance number are issued to identify the affected Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.
System action
The IKE message is ignored; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that it has sent an incorrect message.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
simple_net.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1753I IKE version 1.0 exchange 32 message 0 from 1.2.3.4 port 500 to  5.6.7.8 port 500 is incorrect  
because the message length indicated in the header of the message is too large

EZD1754I Validation failed for COOKIE notify payload received from remote_ip port remote_port to local_ip port local_port

Explanation
When the Internet Key Exchange (IKE) daemon detects a large number of half-open IKE security associations (SAs), it sends a notify payload of type COOKIE to the peer. The peer must duplicate and send back the COOKIE notify payload. The IKE daemon periodically updates the local cookie information and requires that the data in all received COOKIE notify payloads match the local cookie information. If the received cookie data does not match the local cookie information, then it might indicate that the sender is attempting a denial-of-service (DoS) attack against the IKE daemon. This message might be issued with a benign IKE daemon peer if the local cookie information is updated before the peer is able to respond, although such an occurrence is unlikely. A large number of these messages probably indicates a DoS attack.

In the message text:
remote_ip
The remote security endpoint IP specification.

remote_port
The port of the remote IKE daemon peer.

local_ip
The local security endpoint IP specification.

local_port
The port of the local IKE daemon.

System action
IKE daemon processing continues.

Operator response
If more than one EZD1754I message is issued in quick succession, contact the system programmer.

System programmer response
If a large number of EZD1754I messages are issued for the same remote IP address, then the host with that IP address might be mounting a DoS attack against the IKE daemon. Install an IP filter rule to deny IP traffic from that address. If a large number of EZD1754I messages are issued for different remote IP addresses, then it might indicate that an attacker is forging IP addresses. In this case, install IP filter rules to deny IP traffic from all the IP addresses reported as the remote_ip value in the EZD1754I messages. See the information about IP security in z/OS Communications Server: IP Configuration Guide for information about implementing IP filter rules.

User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2SAInitResponse.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1754I Validation failed for COOKIE notify payload received from 1.2.3.4 port 500 to 5.6.7.8 port 500
Explanation

The Internet Key Exchange (IKE) daemon was unable to accept a proposal because the local and remote IKEv2 Security Association (SA) transform attribute values were not the same. IKE daemon processing continues to the next proposal. If no proposals are accepted, the SA negotiation fails. This failure is indicated by message EZD0985I, EZD1021I, or EZD1022I being issued later in syslog.

In the message text:

transform_name
The name of the transform for which the mismatch occurred.

local_attribute_name and remote_attribute_name
The names of the attributes for which the mismatch occurred.

local_attribute_value and remote_attribute_value
The attribute values that were not the same.

local_proposal_number and remote_proposal_number
The proposal numbers.

System action
IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
If the proposal with the mismatch is the one that should be accepted, notify the administrator of the remote and local security endpoints about the mismatch specification and ask that the configuration be updated with correct values. If the proposal with the mismatch is not the one that should be accepted, you can ignore message EZD1755I. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2SAAttribute.cpp

Routing code
11
Automation
This message is output to the syslog.

Example

```
EZD1755I Transform ENCR_AES_CBC Key_Length attribute value 128 in local proposal 1 does not match Key_Length attribute value 256 in remote proposal 1
```

```
EZD1756I The message_type for exchange_type exchange with message ID message_id from remote_ip port remote_port to local_ip port local_port will not be processed because reason
```

Explanation

The message with the specified ID is ignored for the specified reason.

In the message text:

- **message_type**
  The type of message that was ignored. Possible message_type values are request or response.

- **exchange_type**
  The type of exchange as described in RFC 5996 Internet Key Exchange (IKEv2) Protocol. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

- **message_id**
  The message ID as described in RFC 5996 Internet Key Exchange (IKEv2) Protocol. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

- **remote_ip**
  The remote security endpoint IP specification.

- **remote_port**
  The port of the remote Internet Key Exchange (IKE) daemon peer.

- **local_ip**
  The local security endpoint IP specification.

- **local_port**
  The port of the local IKE daemon.

- **reason**
  The description of why the message will not be processed.

System action
The IKE message is ignored; IKE daemon processing continues.

Operator response
None.

System programmer response
Notify the administrator of the remote security endpoint that a protocol error has occurred.

User response
None.
Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2Exchange.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1756I The request for CREATE_CHILD_SA exchange with message ID 2 from 1.2.3.4 port 500 to 5.6.7.8 port 500 will not be processed because the IKE_SA for the message is half-closed

EZD1757I The protection_type payload_type payload is ignored because reason

Explanation
An Internet Key Exchange version 2 (IKEv2) payload is being ignored for the reason given.

In the message text:

protection_type
Specifies whether the payload is protected or unprotected. A protected payload is one that has been decrypted and has passed the integrity check. An unprotected payload is one that was neither encrypted nor checked for integrity.

payload_type
The payload type as described in RFC 5996 Internet Key Exchange (IKEv2) Protocol. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

reason
The reason for ignoring the payload.

System action
The Internet Key Exchange (IKE) message is ignored; IKE daemon processing continues.

Operator response
None.

System programmer response
Notify the administrator of the remote security endpoint that you received an incorrect IKE message.
User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2Request.cpp and IKEv2Response hierarchy.

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1757I The protected SA payload is ignored because it was not expected in this message

EZD1758I Transform transform_name attribute type local_attribute_type in local proposal local_proposal_number does not match attribute type remote_attribute_type in remote proposal remote_proposal_number

Explanation
The Internet Key Exchange (IKE) daemon was unable to accept a proposal because the local and remote IKEv2 Security Association (SA) transform attribute types are not the same. IKE daemon processing continues to the next proposal. If no proposals are accepted, the SA negotiation fails. This failure is indicated by message EZD0985I, EZD1021I, or EZD1022I being issued later in syslog.

In the message text:
transform_name
The name of the transform for which the mismatch occurred.

local_attribute_type and remote_attribute_type
The attribute types that are not the same.

local_proposal_number and remote_proposal_number
The proposal numbers.

System action
IKE daemon processing continues.

Operator response
None.
**System programmer response**

If the proposal with the mismatch is the one that should be accepted, notify the administrator of the remote and local security endpoints about the mismatch specification and ask that the configuration be updated with correct values. If the proposal with the mismatch is not the one that should be accepted, then you can ignore message EZD1758I. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**User response**

None.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

IKEv2SAProposal.cpp

**Routing code**

11

**Descriptor code**

7

**Automation**

This message is output to the syslog.

**Example**

```
EZD1758I Transform ENCR_AES_CBC attribute type 14 in local proposal 1 does not match attribute type 16384 in remote proposal 1
```

```
EZD1759I Transform transform_name attribute type local_attribute_type with local_attribute_format format in local proposal local_proposal_number does not match attribute type remote_attribute_type with remote_attribute_format format in remote proposal remote_proposal_number
```

**Explanation**

The Internet Key Exchange (IKE) daemon was unable to accept a proposal because the local and remote IKEv2 Security Association (SA) transform attribute formats were not the same. IKE daemon processing continues to the next proposal. If no proposals are accepted, the SA negotiation fails. Failure is indicated by message EZD0985I, EZD1021I, or EZD1022I being issued later in syslog.

In the message text:

- `transform_name` The name of the transform for which the mismatch occurred.
- `local_attribute_type` and `remote_attribute_type` The attribute types.
- `local_attribute_format` and `remote_attribute_format` The attribute formats that are not the same.
local_proposal_number and remote_proposal_number

The proposal numbers.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
Notify the administrator of the remote and local security endpoints that a transform attribute was sent in the wrong format.

User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2SAProposal.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example

EZD1759I Transform ENCR_AES_CBC attribute type 14 with fixed-length format in local proposal 1 does not match attribute type 14 with variable-length format in remote proposal 1

EZD1760I Proposal proposal_number contains an unsupported transform ID transform_id for transform type transform_type

Explanation

The Security Association (SA) proposal is not accepted because it contains a transform with an unsupported ID. Internet Key Exchange (IKE) daemon processing continues to the next proposal. If no proposals are accepted, the SA negotiation fails. Failure is indicated by message EZD0985I, EZD1021I, or EZD1022I being issued later in syslog.

In the message text:
**proposal_number**
The number of the proposal that contains the transform with the unsupported ID.

**transform_id**
The transform identifier that is not supported.

**transform_type**
The transform type of the transform with the unsupported ID. The transform types are listed in RFC 5996 *Internet Key Exchange (IKEv2) Protocol*. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

**System action**
If the IKE daemon does not accept any of the proposals, the negotiation fails; IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
Notify the administrator of the remote security endpoint that an SA payload with an unsupported transform ID was received.

**User response**
None.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
IKEv2SATransforms.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to syslog.

**Example**

```
EZD1760I Proposal 3 contains an unsupported transform ID 1 for transform type Encryption Algorithm (ENCR)
```

```
EZD1761I protocol_name protocol proposal proposal_number is incorrectly formatted - reason
```

**1138** z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Explanation
The Security Association (SA) proposal is not accepted because it is not formatted correctly. Internet Key Exchange (IKE) daemon processing continues to the next proposal. If no proposals are accepted, the SA negotiation fails. Failure is indicated by message EZD0985I, EZD1021I, or EZD1022I being issued later in syslog.

In the message text:

**protocol_name**
The name of the protocol.

**proposal_number**
The proposal number within the payload.

**reason**
The reason that the proposal is incorrectly formatted.

System action
If the IKE daemon does not accept any of the proposals, the negotiation fails; IKE daemon processing continues.

Operator response
None.

System programmer response
Notify the administrator of the remote security endpoint that an incorrectly formatted SA proposal was received.

User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2SAProposal.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1761I IKE protocol proposal 1 is incorrectly formatted - proposal too long for payload

EZD1762I In the response SA payload there is more than one transform type transform_type
**Explanation**

Internet Key Exchange version 2 (IKEv2) allows at most one transform of each type in a response Security Association (SA) payload.

In the message text:

*transform_type*

The transform type, described in RFC 5996 *Internet Key Exchange (IKEv2) Protocol*. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

**System action**

The SA negotiation fails; IKE daemon processing continues.

**Operator response**

None.

**System programmer response**

Notify the administrator of the remote security endpoint that an incorrectly formatted SA payload was received in a response.

**User response**

None.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

IKEv2SAInitResponse.cpp IKEv2AuthResponse.cpp IKEv2CreateChildResponse.cpp

**Routing code**

11

**Descriptor code**

7

**Automation**

This message is output to the syslog.

**Example**

```
EZD1762I In the response SA payload there is more than one transform type 2 - Pseudo-random Function (PRF)
```

```
EZD1763I There is more than one proposal in the response SA payload
```
**Explanation**
Internet Key Exchange version 2 (IKEv2) requires that a Security Association (SA) payload in a response include only one proposal.

**System action**
The SA negotiation fails; IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
Notify the administrator of the remote security endpoint that an SA payload with more than one proposal was received in a response.

**User response**
None.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
IKEv2SAInitResponse.cpp IKEv2AuthResponse.cpp IKEv2CreateChildResponse.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to the syslog.

**Example**

EZD1763I There is more than one proposal in the response SA payload

EZD1764I Multiple proposals with proposal number proposal_num were received in the SA payload

**Explanation**
Internet Key Exchange version 2 (IKEv2) Security Association (SA) payloads can contain multiple proposals. The first proposal in an SA payload must be numbered proposal number 1 and each proposal number must be one greater than the previous. This message indicates that the SA payload contained multiple proposals with the same proposal number.

In the message text:
**Proposal number**

The proposal number that appears multiple times in the SA payload.

**System action**

Proposals with the same proposal number are ignored; IKE daemon processing continues.

**Operator response**

None.

**System programmer response**

Notify the administrator of the remote security endpoint that an incorrectly formatted SA payload was received.

**User response**

None.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

IKEv2SAProposal.cpp

**Routing code**

11

**Descriptor code**

7

**Automation**

This message is output to the syslog.

**Example**

EZD1764I Multiple proposals with proposal number 2 were received in the SA payload

EZD1765I Received an IKEv2 notify payload with status type notify_type in exchange_type message_type

**Explanation**

The Internet Key Exchange (IKE) daemon received an IKEv2 notify payload that contains the specified status type. Status type notify payloads convey information about the payload sender or the tunnel, and are not an indication of an error.

In the message text:

notify_type

The decimal value of the notify type field, followed by a short text description of the type as defined in RFC 5996 *Internet Key Exchange (IKEv2) Protocol*. See Appendix A, “Related protocol specifications,” on page 1142.
exchange_type
The type of exchange as described in RFC 5996 Internet Key Exchange (IKEv2) Protocol. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

message_type
The type of the message. Possible message_type values are request or response.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2NotifyPayload.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog

Example
EZD1765I Received an IKEv2 notify payload with status type 40000 (UNKNOWN) in INFORMATIONAL response

EZD1766I The payload_type payload length is reason

Explanation
The specified payload length is incorrect.

In the message text:
payload_type
The payload notation as described in RFC 5996 Internet Key Exchange (IKEv2) Protocol. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs. Possible payload_type values are:

AUTH
An authentication payload

CERT
A certification payload

CERTREQ
A certification request payload

CP
A configuration payload

D
A delete payload

E
An encrypted payload

EAP
An extensible authentication payload

IDi
An initiator identification payload

IDr
A responder identification payload

KE
A key exchange payload

Nonce
A nonce payload

N
A notify payload

SA
A Security Association payload

TSi
An initiator traffic selector payload

TSr
A responder traffic selector payload

V
A vendor identifier payload

Unknown
A payload that is unknown to the Internet Key Exchange (IKE) daemon

reason
The reason why the length is not correct.

System action
The payload is ignored; IKE daemon processing continues.

Operator response
None.

System programmer response
Notify the administrator of the remote security endpoint that you received a payload with an incorrect length.
User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2Message.cpp and IKEv2Payload subclasses

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1766I The TSi payload length is too short for the message

EZD1767I A required **payload_type** payload is missing for a **exchange_type** **message_type**

Explanation
An Internet Key Exchange version 2 (IKEv2) request or response is ignored because it is missing a required payload.

In the message text:

**payload_type**
The payload notation as described in RFC 5996 *Internet Key Exchange (IKEv2) Protocol*. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs. Possible **payload_type** values are:

**AUTH**
An authentication payload

**CERT**
A certification payload

**CERTREQ**
A certification request payload

**CP**
A configuration payload

**D**
A delete payload

**E**
An encrypted payload

**EAP**
An extensible authentication payload
IDI
An initiator identification payload
IDr
A responder identification payload
KE
A key exchange payload
Nonce
A nonce payload
N
A notify payload
SA
A Security Association payload
TSi
An initiator traffic selector payload
TSr
A responder traffic selector payload
V
A vendor identifier payload
Unknown
A payload that is unknown to the IKE daemon

exchange_type
The type of message that contains the missing payload.

message_type
The either request or response.

System action
The Internet Key Exchange (IKE) message is ignored; IKE daemon processing continues.

Operator response
None.

System programmer response
Notify the administrator of the remote security endpoint that you received an incorrect message.

User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2SAInitRequest/Response IKEv2AuthRequest/Response IKEv2CreateChildRequest/Response

Routing code
11

1146  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**Automation**

This message is output to syslog.

**Example**

```
EZD1767I A required SA payload is missing for a IKE_AUTH request
```

**EZD1768I**  
Transform `transform_name` attribute type `local_attribute_type` length `local_attribute_length` in local proposal `local_proposal_number` does not match the length `remote_attribute_length` of attribute type `remote_attribute_type` in remote proposal `remote_proposal_number`.

**Explanation**

The Internet Key Exchange (IKE) daemon was unable to accept a proposal because the local and remote IKEv2 Security Association (SA) transform attribute lengths are not the same. IKE daemon processing continues to the next proposal. If no proposals are accepted, the SA negotiation fails. This failure is indicated by message EZD0985I, EZD1021I, or EZD1022I being issued later in syslog.

In the message text:

- `transform_name`
  The name of the transform for which the mismatch occurred.

- `local_attribute_type` and `remote_attribute_type`
  The attribute types.

- `local_attribute_length` and `remote_attribute_length`
  The attribute lengths that are not the same.

- `local_proposal_number` and `remote_proposal_number`
  The proposal numbers.

**System action**

IKE daemon processing continues.

**Operator response**

None.

**System programmer response**

If the proposal with the mismatch is the one that should be accepted, notify the administrator of the remote and local security endpoints about the mismatch specification and ask that the configuration be updated with correct values. If the proposal with the mismatch is not the one that should be accepted, you can ignore message EZD1768I. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**User response**

None.

**Problem determination**

None.
Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2NotifyPayload.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example

EZD1768I Transform ENCR_AES_CBC attribute type 16384 length 4 in local proposal 1 does not match the length 8 of attribute type 16384 in remote proposal 2

EZD1769I An unsupported transform type transform_type was found in proposal proposal_num

Explanation
The Security Association (SA) proposal with the specified number is not accepted because it contains a transform with an unsupported type. Internet Key Exchange (IKE) daemon processing continues to the next proposal. If no proposals are accepted, the SA negotiation fails. This failure is indicated by message EZD0985I, EZD1021I, or EZD1022I being issued later in syslog.

In the message text:

transform_type
The transform type that is not supported.

proposal_num
The proposal number in the payload.

System action
If the IKE daemon does not accept any of the proposals, the negotiation fails; IKE daemon processing continues.

Operator response
None.

System programmer response
Notify the administrator of the remote security endpoint that an SA payload with an unsupported transform type was received.

User response
None.

Problem determination
None.
Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2SATransform.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1769I An unsupported transform type 6 was found in proposal 2

EZD1770I Transform transform_name value for attribute type local_attribute_type in local proposal local_proposal_number does not match value for attribute type remote_attribute_type in remote proposal remote_proposal_number

Explanation
The Internet Key Exchange (IKE) daemon was unable to accept a proposal because the local and remote IKEv2 Security Association (SA) transform attribute values are not the same. IKE daemon processing continues to the next proposal. If no proposals are accepted, the SA negotiation fails. This failure is indicated by message EZD0985I, EZD1021I, or EZD1022I being issued later in syslog.

In the message text:

transform_name
The name of the transform for which the mismatch occurred.

local_attribute_type and remote_attribute_type
The attribute types.

local_proposal_number and remote_proposal_number
The proposal numbers.

The attribute values are not included in the message because they are variable length and might be as large as 64 kilobytes of data. To obtain the remote proposal value, activate the formatting of IKE messages by using IKE syslog level 8 and attempt the SA negotiation.

System action
IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
If the proposal that contains the mismatch is the one that should be accepted, either alter the local policy to accept the value in this proposal or notify the administrator of the remote security endpoint about the mismatch and ask the administrator to alter the remote configuration to use the correct values. See the information about
Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy and the IkeSyslogLevel statement.

User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2SAAttribute.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example

```
EZD1770I Transform ENCR_AES_CBC value for attribute type 16384 in local proposal 1 does not match value for attribute type 16384 in remote proposal 2
```

```
EZD1771I   IKE version version Security Association sa_generation for tunnel tunnel_id rekeyed
```

Explanation
This message indicates that a Security Association (SA) was rekeyed. Internet Key Exchange (IKE), Encapsulated Security Payload (ESP), and Authentication Header (AH) SAs use secret keys that should be used only for a limited amount of time and to protect a limited amount of data. Rekeying is the reestablishment of SAs to take the place of ones that expire.

Additional diagnostic messages with the same message instance number are issued to identify the affected SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:

**version**
The version of the IKE protocol for the SA that was rekeyed.

**sa_generation**
The number used to differentiate SAs for the same tunnel. The first SA created for a tunnel is number 1.

**tunnel_id**
The tunnel prefix and number used to identify the tunnel. The tunnel prefix is K for an IKE tunnel and Y for a dynamic tunnel.
System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2IKESA.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example

EZD1771I IKE version 1.0 Security Association 2 for tunnel Y17 rekeyed

EZD1772I IKE version version Security Association sa_generation for tunnel tunnel_id reauthenticated

Explanation
This message indicates that a Security Association (SA) was reauthenticated.
Additional diagnostic messages with the same message instance number are issued to identify the affected SA.
The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.
In the message text:

version
The version of the IKE protocol for the SA that was reauthenticated.

sa_generation
The number used to differentiate SAs for the same tunnel. The first SA created for a tunnel is number 1.
**tunnel_id**

The tunnel prefix and number used to identify the tunnel. The tunnel prefix is K for an IKE tunnel and Y for a dynamic tunnel.

**System action**
IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
IKEv2IKESA.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to the syslog.

**Example**

```
EZD1772I IKE version 1.0 Security Association 2 for tunnel K11 reauthenticated
EZD1773I Received an IKEv2 notify payload with error type notify_type in exchange_type message_type
```

**Explanation**
The Internet Key Exchange (IKE) daemon received an IKEv2 notify payload that contains an error of the specified type. Additional diagnostic messages that have the same message instance number are issued to identify the affected SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:
**notify_type**
The decimal value of the notify type field, followed by a short text description of the type as defined in RFC 5996 *Internet Key Exchange (IKEv2) Protocol*. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs. If the decimal value received is not a known error type, the text description is **UNKNOWN**.

**exchange_type**
The type of exchange as described in RFC 5996 *Internet Key Exchange (IKEv2) Protocol*. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

**message_type**
The type of the message. Possible *message_type* values are request or response.

**System action**
IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
Use the *notify_type* value to identify the error. For some error types, you might have to make changes to the security policy, or request that the peer IKE node make changes to its policy, to resolve the issue. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**User response**
None.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
IKEv2Response.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to syslog.

**Example**

```
EZD1773I Received an IKEv2 notify payload with error type 17 (INVALID_KE_PAYLOAD) in IKE_SA_INIT response
```

```
EZD1774I IKEv2 payload type payload_type was received but was ignored because it is not supported
```
Explanation
An Internet Key Exchange version 2 (IKEv2) payload type that is not supported was received, but it was ignored.

In the message text:

payload_type
The payload type in the Internet Key Exchange (IKE) message that is not supported. IKEv2 payload types are described in RFC 5996 *Internet Key Exchange (IKEv2) Protocol*. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs. The payload types that are supported are described in *z/OS Communications Server: IP Configuration Guide*.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
Notify the administrator of the remote security endpoint that a payload type is being ignored.

User response
None.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2Payload.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1774I IKEv2 payload type 5 was received but was ignored because it is not supported

EZD1775I IKE version version Security Association sa_generation for tunnel tunnel_id created for protecting proto_name traffic between local_ip local_selector_type local_selectors and remote_ip remote_selector_type remote_selectors

1154 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Explanation
A new Security Association (SA) has been created with the characteristics given.

In the message text:

**version**
The IKE protocol version used to create the SA.

**sa_generation**
The number used to differentiate SAs for the same tunnel. The first SA created for a given tunnel is number 1.

**tunnel_id**
The tunnel prefix and number used to identify the tunnel. The tunnel prefix is K for an IKE tunnel and Y for a dynamic tunnel.

**proto_name**
The protocol of the traffic protected by the tunnel.

**local_ip**
The IP address of the local traffic protected by the tunnel.

**local_selector_type and remote_selector_type**
The type of upper-layer selectors protected by the tunnel.

**local_selectors**
The upper-layer selectors of the local traffic protected by the tunnel.

**remote_ip**
The IP address of the remote traffic protected by the tunnel.

**remote_selectors**
The upper-layer selectors of the remote traffic protected by the tunnel.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2SecurityAssociation.cpp

Routing code
11
**Descriptor code**

7

**Automation**

This message is output to the syslog.

**Example**

<table>
<thead>
<tr>
<th>Description</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1775I IKE version 1.0 Security Association 1 for tunnel K1 created for</td>
<td>protecting IKE traffic between 10.11.5.4 port 500 and 10.11.4.5 port 500</td>
</tr>
<tr>
<td>EZD1775I IKE version 1.0 Security Association 1 for tunnel Y2 created for</td>
<td>protecting ALL(0) traffic between 10.81.2.1 N/A N/A and 10.81.8.1 N/A N/A</td>
</tr>
</tbody>
</table>

**EZD1776I IKE version version Security Association sa_generation for tunnel tunnel_id deleted**

**Explanation**

This message indicates that a Security Association (SA) was deleted. Multiple SAs are created and deleted to carry tunnel traffic. The deletion of an SA does not imply that the tunnel has ended or is unavailable.

In the message text:

- **version**
  - The IKE protocol version use to delete the SA.

- **sa_generation**
  - The number used to differentiate SAs for the same tunnel. The first SA created for a tunnel is number 1.

- **tunnel_id**
  - The tunnel prefix and number used to identify the tunnel. The tunnel prefix is K for an IKE tunnel and Y for a dynamic tunnel.

**System action**

IKE daemon processing continues.

**Operator response**

None.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

CommonIPsecSA.cpp, CommonIKESA.cpp
Example

EZD1776I IKE version 1.0 Security Association 1 for tunnel K1 deleted

EZD1777I IKE version version Security Association sa_generation for tunnel tunnel_id has the following attributes - identities : id_protection local authentication : local_auth_method remote authentication : remote_auth_method encryption : encr_function integrity : integ_function PRF : psuedo_random_function DH : dh_group_name lifetime : lifetime lifesize : lifesize

Explanation

This message displays the attributes of a new IKE Security Association (SA).

In the message text:

**version**
The version of the IKE protocol that was used to create the SA.

**sa_generation**
The number used to differentiate SAs for the same tunnel. The first SA created for a tunnel is number 1.

**tunnel_id**
The tunnel prefix and number used to identify the IKE tunnel. The tunnel prefix is K.

**id_protection**
Indicates whether the confidentiality of the local and remote IKE identities is ensured for the IKE tunnel.

**local_auth_method**
The method that the remote peer must use to authenticate the local endpoint.

**remote_auth_method**
The method that IKED must use to authenticate the remote endpoint.

**encr_function**
The name of the encryption function that the IKE tunnel used to provide data confidentiality.

**integ_function**
The name of the integrity function that the IKE tunnel used to provide data integrity.

**psuedo_random_function**
The name of the pseudo-random function that was used to seed the keying material.

**dh_group_name**
The Diffie-Hellman group ID that was used to generate the keying material.

**lifetime**
The length of time that the SA lives in seconds.

**lifesize**
The life size of the SA in kilobytes.

System action
IKE daemon processing continues.
Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2IKESA.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1777I  IKE version 1.0 Security Association 1 for tunnel K1 has the following attributes -
identities : unprotected local authentication : PresharedKey  remote authentication :
PresharedKey encryption : AES-CBC integrity : HMAC-SHA1  PRF : PRF_HMAC_SHA1 DH : Group5
lifetime : 28800 lifesize : NONE

EZD1778I  exchange_name message_type message ID msg_id replay detected from remote_ip port
remote_port to local_ip port local_port

Explanation
The Internet Key Exchange (IKE) daemon received a message that is a copy of a message that was received and processed earlier. The previously received and processed messages (replays) are ignored.

In the message text:
exchange_name
The name of the exchange for the message.
message_type
The message type.
msg_id
The message ID number.
remote_ip
The remote security endpoint IP specification.
**remote_port**
The remote port of the IKE daemon peer.

**local_ip**
The local security endpoint IP specification.

**local_port**
The local port of the IKE daemon.

**System action**
The IKE message is not processed; IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
Replays are not errors. Replays might be caused by lost packets or network congestion. Usually, no action is required. If there are a large number of replays, notify the administrator of the remote security endpoint that you are receiving a large number of replays.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
IKEv2ExchangeList.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to syslog.

**Example**
```
EZD1778I IKE_SA_INIT request message ID 0 replay detected from 1.2.3.4 port 500 to 5.6.7.8 port 500
```

```
EZD1779I IKE version version Security Association sa_generation for tunnel tunnel_id has the following attributes - encapsulation : encap_mode encryption : encr_function integrity : integ_function lifetime : lifetime lifesize : lifesize VpnLife : vpn_life PFS : dh_group_name
```

**Explanation**
A new Security Association (SA) has been created with the specified characteristics.
In the message text:

**version**
The IKE protocol version used to create the SA.

**sa_generation**
The number used to differentiate SAs for the same tunnel. The first SA created for a given tunnel is number 1.

**tunnel_id**
The tunnel prefix and number used to identify the dynamic tunnel. The tunnel prefix is Y.

**encap_mode**
The tunnel encapsulation mode, which is either TUNNEL or TRANSPORT.

**encr_function**
The name of the encryption function used by the dynamic tunnel to provide data confidentiality.

**integ_function**
The name of the integrity function used by the dynamic tunnel to provide data integrity.

**lifetime**
The length of time that the SA lives in seconds.

**lifsize**
The SA lifesize, in kilobytes.

**vpn_life**
Specifies how long IPsec SAs should continue to be rekeyed, in seconds. The *vpn_life* value is set for a dynamic tunnel when the first SA is established for the tunnel.

**dh_group_name**
The Diffie-Hellman (DH) group ID used for perfect forward secrecy. For IKE version 2.0, the first IPsec SA created under each IKE SA uses the DH group that is configured for the IKE SA, regardless of what was configured in the policy for that IPsec SA. In all other cases, the DH group used is the one that is configured in the policy for the IPsec SA.

**System action**
IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
IKEv2ChildSA.cpp
Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example

EZD1779I IKE version 1.0 Security Association 1 for tunnel Y2 has the following attributes -
capsulation : TRANSPORT encryption : DES_CBC_8 integrity : HMAC_SHA1 lifetime : 14400
lifsize : NONE VpnLife : NONE PFS : GROUP1

EZD1780I The message_type for exchange_type exchange with message ID message_id from remote_ip
port remote_port to local_ip port local_port is outside the expected window of
lowest_expected to highest_expected

Explanation
The message in the exchange with the specified ID is ignored because the ID is not in the expected message ID
window.

In the message text:

message_type
   The type of message. Possible message_type values are request or response.

exchange_type
   The type of exchange as described in RFC 5996 Internet Key Exchange (IKEv2) Protocol. See Appendix A,
   "Related protocol specifications," on page 1365 for information about accessing RFCs.

message_id
   The ID of the message.

remote_ip
   The remote security endpoint IP specification.

remote_port
   The remote port of the Internet Key Exchange (IKE) daemon peer.

local_ip
   The local security endpoint IP specification.

local_port
   The local port of the IKE daemon.

lowest_expected
   The lowest message ID number that is expected from the remote security endpoint.

highest_expected
   The highest message ID number that is expected from the remote security endpoint.

System action
The IKE message is not processed; IKE daemon processing continues.

Operator response
None.
**System programmer response**
Notify the administrator of the remote security endpoint that a protocol error occurred.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
IKEv2ExchangeList.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to the syslog.

**Example**
```
EZD1780I The request for IKE_AUTH exchange with message ID 2 from 1.2.3.4 port 500 to 5.6.7.8 port 500 is outside the expected window of 1 to 1
```

EZD1781I Received a delete payload with an unrecognized SPI value for protocol protocol

**Explanation**
The Internet Key Exchange (IKE) daemon received a delete payload command from an IKE peer that contains an unrecognized security parameter index (SPI) value that represents an IPsec tunnel that is to be deleted. This situation might occur if IKED was restarted, if a sysplex-wide Security Association (SWSA) takeover occurred recently, or if packets are being dropped in the network.

In the message text:

**value**
The hexadecimal SPI value that was received.

**protocol**
The name of the protocol. Possible protocol values are AH or ESP.

**System action**
The SPI that is to be deleted is ignored; IKE daemon processing continues.

**Operator response**
Contact the system programmer.
**System programmer response**

Compare local IKE syslog output with any log information available at the remote IKE security endpoint to determine whether the SPI value should have been known to IKE.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

IKEv2IKESA.cpp infoXchng.cpp

**Routing code**

11

**Descriptor code**

7

**Automation**

This message is output to the syslog.

**Example**

```
EZD1781I Received a delete payload with an unrecognized SPI 44BA7983 for protocol ESP
```

```
EZD1782I Received a delete payload with SPI spi_value for protocol protocol that does not belong to this IKE_SA
```

**Explanation**

The Internet Key Exchange (IKE) daemon received a message from an IKEv2 peer that contained a security parameter index value that represents a child Security Association (SA) that is to be deleted. However, the message was protected by an IKE SA other than the IKE SA to which the child SA belongs. The IKEv2 peer is in error; RFC 5996 *Internet Key Exchange (IKEv2) Protocol* section 1.4 requires that notification messages for child SAs are to be protected only by the IKE SA that generated the child SA. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

In the message text:

- **spi_value**
  The hexadecimal SPI value that was received.

- **protocol**
  The protocol value. Possible protocol values are AH or ESP.

**System action**

The tunnel to be deleted is ignored; IKE daemon processing continues.
Operator response
Contact the system programmer.

System programmer response
Examine the logging information that is available at the remote IKE security endpoint to determine whether the child SA was incorrectly deleted.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2IKESA.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1782I Received a delete payload with SPI 44BA7983 for protocol ESP that does not belong to this IKE_SA

EZD1783I Received a notify payload with an unrecognized SPI value for protocol protocol

Explanation
The Internet Key Exchange (IKE) daemon received a notify payload command from an IKE peer that contains an unrecognized security parameter index (SPI) value that represents an IPsec tunnel. This situation might occur if IKED was restarted, if a sysplex wide Security Association (SWSA) takeover occurred recently, or if packets are being dropped in the network.

In the message text:

value
The hexadecimal SPI value that was received.

protocol
The name of the protocol. Possible protocol values are AH or ESP.

System action
The notify payload is ignored; IKE daemon processing continues.
Operator response
Contact the systems programmer.

System programmer response
Compare IKE syslog output with any log information available at the remote IKE security endpoint to determine whether the SPI value should have been known to IKE.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
infoXchg.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1783I Received a notify payload with an unrecognized SPI 44BA7983 for protocol ESP

EZD1784I Received optional IDr payload but could not find applicable KeyExchangeRule - LocalIp: LSIP RemoteIp: RSIP LocalID: LSID RemoteID: RSID

Explanation
The remote IKEv2 peer provided an optional Identification - Responder (IDr) payload in its IKE_SA_INIT request to the local Internet Key Exchange (IKE) daemon. This IDr payload contained an identity that did not match the local key exchange policy, so the local identity (LocalID) that is specified by the optional IDr payload is ignored.

Additional messages that have the same message instance number are issued to identify the affected SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:

**LSIP**
- The local security endpoint IP address.

**RSIP**
- The remote security endpoint IP address.
**LSID**
The local security endpoint identity as provided by the remote IKEv2 peer. The LSID is an ID type followed by optional data.

**RSID**
The remote security endpoint identity as provided by the remote IKEv2 peer. The RSID is an ID type followed by optional data.

The ID type is one of the values defined in RFC 5996 *Internet Key Exchange (IKEv2) Protocol* section 1.4. For example, `ID_IPV4_ADDR`, `ID_FQDN`, or `ID_IPV6_ADDR`

See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

**System action**
IKE daemon processing continues; the local identity (LocalID) that is specified by the optional IDr payload is ignored.

**Operator response**
Add a suitable KeyExchangeRule statement for the classification to the IPSec policy, if necessary. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
policymgr.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to syslog.

**Example**

```
EZD1784I Received optional IDr payload but could not find applicable KeyExchangeRule -
LocalIp : 9.1.1.1 RemoteIp : 9.2.2.2 LocalID : ID_FQDN example.ibm.com
RemoteID : ID_IPV4_ADDR 9.2.2.2
```

1166  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
EZD1785I  Tentative KeyExchangeRule rule1 replaced with final KeyExchangeRule rule2

Explanation
When an IKE negotiation is started, the KeyExchangePolicy statement is searched to locate a matching KeyExchangeRule statement for the negotiation. When this KeyExchangeRule statement is located, the local and remote security endpoint identities are not known with certainty, so this KeyExchangeRule statement is considered tentative until the local and remote identities are known. Then a new search for a KeyExchangeRule statement is performed to locate the final rule.

The policy on the final KeyExchangeRule statement must be consistent with the policy chosen for the Security Association. For IKEv1, see the information about ISAKMP main mode limitations in z/OS Communications Server: IP Diagnosis Guide for more information about IKEv1. For IKEv2, see the information about key exchange limitations in z/OS Communications Server: IP Diagnosis Guide for more information about IKEv2.

In the message text:

rule
The name of the tentative KeyExchangeRule statement.

rule2
The name of the final KeyExchangeRule statement.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
CommonDomainOfInterpretation.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.
EZD1786I Cannot negotiate IKEv1 tunnel for filter rule rulename using specific types or codes

Explanation
IKEv1 does not support the negotiation of tunnels for specific ICMP types and codes, ICMPv6 types and codes, or MIPv6 types and codes. The filter rule specified by the rulename value uses an IpService definition that specifies either individual types and codes or a range of types and codes, for ICMP, ICMPv6, or MIPv6 traffic. A tunnel negotiation was attempted that used the specified filter rule with a KeyExchangeAction statement that requires IKEv1 to initiate the negotiation. Tunnels that use either individual types and codes or a range of types and codes can be negotiated for IKEv2, but not for IKEv1.

In the message text:

rulename
The name of the filter rule.

System action
The tunnel activation fails; IKE daemon processing continues.

Operator response
None.

System programmer response
Examine the policy definition for the rule name and modify it to correct the error. Potential changes include:
- Specify HowToInitiateVersion IKEv2.
- Do not code specific values or ranges of ICMP, ICMPv6, or MIPv6 types and codes.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
cfg Adapter.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.
Example

EZD1786I  Cannot negotiate IKEv1 tunnel for filter rule TimestampRequestReply using specific types or codes

EZD1787I  Received unsupported IKEv2 traffic selector specification

Explanation
The Internet Key Exchange (IKE) daemon does not support a traffic selector specification that it received from an IKEv2 peer. The IKE daemon does not support the following types of traffic selectors:

• Traffic selectors that use a distinct set of port values instead of a contiguous range; these traffic selectors are called disjoint traffic selectors. If the peer is acting as an initiator, the IKE daemon attempts to find a pairing of its proposed traffic selectors that is not disjoint. However, if the peer is acting as a responder, the IKE daemon cannot accept a counter-proposal that contains disjoint traffic selectors, so it fails the tunnel activation.

• Traffic selectors that are asymmetrical; for example, traffic selectors that contain ICMP type 13 in one direction, but ICMP type 14 in the other direction. If the peer is acting as an initiator, the IKE daemon attempts to find a pairing of its proposed traffic selectors that is not asymmetrical. If the peer is acting as a responder and its returned traffic selectors are asymmetrical, the IKE daemon fails the tunnel activation.

• Traffic selectors that contain port, type, or code specifications for any protocol other than TCP, UDP, ICMP, ICMPv6, or MIPv6. RFC 5996 Internet Key Exchange (IKEv2) Protocol makes provisions for negotiating port, type, and code values for these protocols. If the peer is acting as an initiator, the IKE daemon attempts to find a pairing of its proposed traffic selectors that has recognizable port, type, and code specifications. If the peer is acting as a responder and its returned traffic selectors contain any unrecognized port, type, or code specifications, the IKE daemon fails the tunnel activation. See Appendix A, “Related protocol specifications,” on page 1365 for information about accessing RFCs.

To perform further diagnosis, enable the formatted packet trace option in your IkeSyslogLevel configuration settings, and retry tunnel activation. Your syslog contains the exact traffic selector values that the peer is proposing.

System action
The tunnel activation fails; IKE daemon processing continues.

Operator response
None.

System programmer response
Contact the administrator of the remote IKE peer node to modify their policies so that the resulting traffic selectors are compatible with the restrictions listed above.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2DomainOfInterpretation.cpp
Routing code
11

Descriptor code
7

Automation
This message is output to syslog

Example
EZD1787I Received unsupported IKEv2 traffic selector specification

EZD1788I Received an IKEv2 traffic selector in a response that does not match the local proposal

Explanation
The Internet Key Exchange (IKE) daemon proposed a traffic selector specification to the IKEv2 peer, but received a response from the IKEv2 peer whose traffic selectors do not match the proposal that was sent to the peer.

If you enable the formatted packet trace option in your IkeaSyslogLevel configuration settings, your syslog contains the exact traffic selector values that the peer is sending for further diagnosis.

System action
The tunnel activation fails; IKE daemon processing continues.

Operator response
None.

System programmer response
Contact the administrator of the remote IKE peer node to modify their policies so that the resulting traffic selectors are compatible with the local proposals.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2DomainOfInterpretation.cpp

Routing code
11
**Automation**
This message is output to syslog

**Example**
```
EZD1788I Received an IKEv2 traffic selector in a response that does not match the local proposal
```

**EZD1789I** The remote security endpoint `requested_ep` is not included within the address `rule_ep` taken from `rule_name`

**Explanation**
During the negotiation of a tunnel-mode Security Association (SA) the IKE daemon determined that the requested IP addresses are not included in the security endpoint that the IKE daemon chose from the policy rule or destination address.

In the message text:
- `requested_ep` The IP address or IP address range endpoint that the endpoint requested for the tunnel-mode SA. The `requested_ep` value is the value configured for the InitiateToLocation parameter on the IpLocalStartAction statement for the applicable filter rule, or if the InitiateToLocation parameter was not configured, the value is the destination data address for the tunnel activation.
- `rule_ep` The remote security endpoint IP address or IP address range that is configured on the IpLocalStartAction statement for the applicable filter rule.
- `rule_name` The name of the IpFilterRule statement that is used for the tunnel activation. The IpFilterRule statement refers to an IpLocalStartAction statement that specifies a RemoteSecurityEndpoint parameter with the indicated `rule_ep` value.

**System action**
The SA negotiation fails; IKE daemon processing continues.

**Operator response**
Contact the systems programmer. Problem Determination: Not applicable.

**System programmer response**
Alter the local policy configuration to specify the following:
- An InitiateToLocation value on the applicable IpLocalStartAction statement; the InitiateToLocation value must be within the range of the RemoteSecurityEndpoint location value.
- A RemoteSecurityEndpoint parameter on the applicable IpLocalStartAction statement; the RemoteSecurityEndpoint parameter must encompass the actual remote security endpoint address.

See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**User response**
Not applicable.
Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
policymgr.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example
EZD1789I The remote security endpoint 10.11.4.5 is not included within the address 10.8.5.5/24 taken from S4-S5_IKE1311-5A

EZD1790I Error event IKE version Security Association sa_generation for tunnel tunnel_id

Explanation
This message indicates that an error occurred while the Internet Key Exchange (IKE) daemon was rekeying or reauthenticating a Security Association (SA).

Additional messages that have the same message instance number are issued to identify the affected SA. The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:

**event**
The event that encountered the error. Possible event values are rekeying or reauthenticating.

**version**
The version of the IKE protocol for the SA.

**sa_generation**
The number used to differentiate SAs for the same tunnel. The first SA that is created for a tunnel is number 1.

**tunnel_id**
The tunnel prefix and number used to identify the tunnel. The tunnel prefix is K for an IKE tunnel or Y for a dynamic tunnel.

System action
The reauthentication or rekeying fails. The existing SA might be expired. IKE daemon processing continues.

Operator response
None.
**System programmer response**
Consult the syslog output to identify other messages that indicate the cause of the error.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
IKEv2IKESA.cpp, IKEv2ChildSA.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to syslog.

**Example**

```
EZD1790I Error reauthenticating IKE version 2.0 Security Association 1 for tunnel K3
```

**Explanation**
This message indicates that an attempt to reauthenticate or rekey a Security Association (SA) will not be made because the SA is not being used.

In the message text:

- **version**
  The version of the Internet Key Exchange (IKE) protocol for the SA.

- **sa_generation**
  The number used to differentiate SAs for the same tunnel. The first SA that is created for a tunnel is number 1.

- **tunnel_id**
  The tunnel prefix and number used to identify the tunnel. The tunnel prefix is K for an IKE tunnel and Y for a dynamic tunnel.

- **action**
  The action that is being skipped. Possible action values are rekeyed or reauthenticated.

**System action**
IKE daemon processing continues.
**Example**

```
EZD1791I  IKE version 2.0 Security Association 1 for tunnel K3 will not be reauthenticated

EZD1792I  IKE version version Security Association phase2_generation for tunnel phase2_tunnel_id rekeyed due to reauthentication of Security Association phase1_generation for tunnel phase1_tunnel_id
```

**Explanation**

This message indicates that a phase 2 Security Association (SA) was rekeyed because its associated phase 1 SA was reauthenticated. SAs use secret keys that should be used only for a limited amount of time and to protect a limited amount of data. Rekeying is the reestablishment of SAs to take the place of ones that expire. When a phase 1 SA is reauthenticated, all of its associated phase 2 SAs are rekeyed.

Additional messages that have the same message instance number are issued to identify the affected SA. The message instance number precedes the message number in the log output and is used to group related messages from the Internet Key Exchange (IKE) daemon.

In the message text:

- **version**
  - The version of the IKE protocol for the SA that was rekeyed.

- **phase2_generation**
  - The number used to differentiate SAs for the same tunnel. The first SA that is created for a tunnel is number 1.
**phase2_tunnel_id**
The tunnel prefix and number used to identify a phase 2 tunnel. The tunnel prefix is Y.

**phase1_generation**
The number used to differentiate SAs for the same tunnel. The first SA that is created for a tunnel is number 1.

**phase1_tunnel_id**
The tunnel prefix and number used to identify a phase 1 tunnel. The tunnel prefix is K.

**System action**
IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: Network Security Server

**Module**
CommonIPsecSA.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to syslog.

**Example**

<table>
<thead>
<tr>
<th>EZD1792I</th>
<th>IKE version 2.0 Security Association 2 for tunnel Y8 rekeyed due to reauthentication of security association 2 for tunnel K3</th>
</tr>
</thead>
</table>

**EZD1793I** Obsolete IKE configuration file parameter param on line linenum is ignored

**Explanation**
The specified IKE configuration file parameter is obsolete. The IKE daemon ignores the parameter and any values that are specified with the parameter.
In the message text:

**param**
The obsolete parameter that was specified.

**linenum**
The line number on which the obsolete parameter was specified.

**System action**
The IKE daemon continues processing.

**Operator response**
Contact the system programmer.

**System programmer response**
Remove the obsolete parameter from the IKE configuration file.

**User response**
Not applicable.

**Problem determination**
None

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
ike_config.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to the syslog.

**Example**

```
EZD1793I  Obsolete IKE configuration file parameter KeyRetries on line 35 is ignored
```

```
EZD1794I  Local activation of a dynamic tunnel failed for proto_name traffic between local_ip local_selector_type local_selector and remote_ip remote_selector_type remote_selector
```

**Explanation**
A local activation of a dynamic tunnel for protecting traffic with the characteristics given failed.

Additional diagnostic messages with the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.
In the message text:

**proto_name**
The protocol of the traffic to be protected by the tunnel.

**local_ip**
The local IP address of the traffic to be protected by the tunnel.

**local_selector_type and remote_selector_type**
The type of upper-layer selectors to be protected by the tunnel. The selector type value N/A implies that a selector type is not applicable.

**local_selector**
The upper-layer selector of the local traffic to be protected by the tunnel. The selector value N/A implies that a local selector is not applicable.

**remote_ip**
The remote IP address of the traffic to be protected by the tunnel.

**remote_selector**
The upper-layer selector of the remote traffic to be protected by the tunnel. The selector value N/A implies that a remote selector is not applicable.

**System action**
The SA negotiation fails; IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
Review additional diagnostic messages to determine the cause of the failure. After resolving the problem, attempt the tunnel activation again.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
anchor_ureq.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to the syslog.
Example

EZD1794I Local activation of a dynamic tunnel failed for UDP(17) traffic between 1.2.0.1 port 2000 and 1.1.0.1 port 3000
EZD1794I Local activation of a dynamic tunnel failed for IP(4) traffic between 1.2.0.1 N/A N/A and 1.1.0.1 N/A N/A

EZD1795I A matching IpFilterRule with an IpDynVpnAction was not found for protecting proto_name traffic between local_ip local_selector_type local_selector and remote_ip remote_selector_type remote_selector

Explanation

An IKE negotiation failed because a matching IpFilterRule statement could not be found or because the IpFilterRule statement that was found did not have an associated IpDynVpnAction statement in the policy agent configuration file.

When the connectivity rules in the GUI are configured with the IBM Configuration Assistant for z/OS Communications Server, they correspond to the policy agent configuration IpFilterRule statements. The security levels that use dynamic tunnels in the GUI correspond to the IpDynVpnAction statements.

In the message text:

proto_name
The protocol of the traffic to be protected by the tunnel.

local_ip
The local IP address of the traffic to be protected by the tunnel.

local_selector_type and remote_selector_type
The type of upper-layer selectors to be protected by the tunnel. The selector type value N/A means that the selector type is not applicable.

local_selector
The upper-layer selector of the local traffic to be protected by the tunnel. The local selector value N/A means that the local selector is not applicable.

remote_ip
The remote IP address of the traffic to be protected by the tunnel.

remote_selector
The upper-layer selector of the remote traffic to be protected by the tunnel. The remote selector value N/A means that the remote selector is not applicable.

System action
The Security Association (SA) negotiation fails; IKE daemon processing continues.

Operator response
None.

System programmer response
If the specified traffic is to be protected by a dynamic SA, then update the configuration. If the remote system is behind a NAT, ensure that the remote_ip in the filter rule is the public address of the peer system. If the remote system is behind a gateway behind a NAT, ensure the remote_ip in the filter rule is the public address of the gateway.

If you are updating the configuration without the IBM Configuration Assistant for z/OS Communications Server, update the IpFilterPolicy statement to define an IpFilterRule statement with an IpDynVpnAction statement for the traffic pattern identified in the message. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.
If you are updating the configuration with the IBM Configuration Assistant for z/OS Communications Server, update the TCP/IP stack connectivity rules so that the specified traffic is protected by a security level that uses a dynamic tunnel. See the online help in the GUI for additional information.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
CommonIPsecSA.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to the syslog.

Example

<table>
<thead>
<tr>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1795I</td>
<td>A matching IpFilterRule with an IpDynVpnAction was not found for protecting UDP(17) traffic between 1.2.0.1 port 2000 and 1.1.0.1 port 3000</td>
</tr>
<tr>
<td>EZD1795I</td>
<td>A matching IpFilterRule with an IpDynVpnAction was not found for protecting IP(4) traffic between 1.2.0.1 N/A N/A and 1.1.0.1 N/A N/A</td>
</tr>
</tbody>
</table>

EZD1796I  Simultaneous rekeying of IKE version **version** Security Association **sa_generation** for tunnel **tunnel_id** detected

Explanation
An attempt by both security endpoints to simultaneously rekey the same Security Association has been detected.

In the message text:

**version**
The version of the Internet Key Exchange (IKE) protocol for the SA.

**sa_generation**
The number used to differentiate SAs for the same tunnel. The first SA that is created for a tunnel is number 1.

**tunnel_id**
The tunnel prefix and number used to identify the tunnel. The tunnel prefix is K for an IKE tunnel and Y for a dynamic tunnel.

System action
The IKE daemon processing continues.
Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2CreateChildRequest.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1796I Simultaneous rekeying of IKE version 2.0 Security Association 1 for tunnel K3 detected

EZD1797I Traffic specification requires NON_FIRST_FRAGMENTS_ALSO but IKEv2 peer did not send it

Explanation
When an IP packet that has upper-layer transport selectors (TCP port, UDP port, ICMP type and code, or MIPv6 type) is fragmented, only the first fragment contains the transport selectors. The remaining fragments are known as non-first fragments. There are potential security risks when you filter these non-first fragments because the port, type or code values are unknown. Because of these risks, RFC 5996 Internet Key Exchange (IKEv2) Protocol requires IKEv2 peers to use the NON_FIRST_FRAGMENTS_ALSO notify payload to negotiate support for non-first fragments. This negotiation determines whether non-first fragments are allowed to be carried on the IPSec Security Association (SA). They are allowed if the SA meets the following criteria:

- The SA is using tunnel mode rather than transport mode.
- The SA applies to TCP, UDP, ICMP, ICMPv6, or MIPv6 traffic and has a port, type or code, specification other than ALL.
- The SA endpoints support stateful fragment checking, or the z/OS end of the SA carries only local traffic. Local traffic is filtered before it is fragmented, so it is not a security risk.

z/OS Communications Server does not implement stateful fragment checking, so it does not require the NON_FIRST_FRAGMENTS_ALSO notify payload for SAs that are carrying routed traffic. However, z/OS Communications Server does require the NON_FIRST_FRAGMENTS_ALSO notify payload for SAs that are
carrying local traffic because it sends local non-first fragments over the same SA as the first fragments. If the peer does not include this notify payload, it cannot receive the non-first fragments that the z/OS might send over this SA. z/OS will fail the SA negotiation and generate this message, because the peer is not prepared to receive all possible traffic that z/OS Communications Server might send over the SA.

**System action**
The SA negotiation fails. The IKE daemon processing continues.

**Operator response**
None.

**System programmer response**
Consult the syslog output to identify other messages that indicate which policy rules relate to the error. To prevent this failure, perform one of the following actions:

- Configure the remote security endpoint to enable stateful fragment checking.
- Configure the SA at both security endpoints to use transport mode rather than tunnel mode.
- Configure the SA at both security endpoints to cover all ports, types, or codes rather than to cover specific ports, types, or codes. To ensure that the SA covers all ports, types, or codes, configure both the IP filter rule and the granularity settings at both security endpoints. The IP filter rule must specify all ports, types, or codes. The on-demand granularity settings for port, type, and code must be set to use the values defined in the IP filter rule rather than the on-demand packet. For z/OS Communications Server, the IP filter rule selectors are configured on the IpService statement and granularity settings are configured on the IpLocalStartAction statement. See the information about the IpService statement and the IpLocalStartAction statement in z/OS Communications Server: IP Configuration Reference.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
IKEv2TSRequest.cpp, IKEv2TSResponse.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to syslog.
EZD1797I  Traffic specification requires NON_FIRST_FRAGMENTS_ALSO butIKEv2 peer did not send it

EZD1798I  *PolicySource policy* requires the SharedKey parameter but none is specified in *KeyExchangeRule KERname*

**Explanation**

The associated KeyExchangeRule statement for this Internet Key Exchange (IKE) Security Association (SA) negotiation did not specify a shared key to be used in the negotiation. Because a pre-shared key is required and none was given, the negotiation failed.

In the message text:

*PolicySource*  
The *PolicySource* value is either local or remote.

*KERname*  
The KeyExchangeRule name configured in the policy.

**System action**

The IKE SA negotiation fails; IKE daemon processing continues.

**Operator response**

Contact the system programmer.

**System programmer response**

Either specify a preshared key to be used in this negotiation on the SharedKey parameter in the KeyExchangeRule statement or specify the use of digital signature authentication. If the PolicySource value is local, you can specify a digital signature authentication method on the HowToAuthMe keyword on the KeyExchangeAction statement that is associated with the specified KeyExchangeRule. If PolicySource is remote and digital signature authentication is preferred to specifying a shared key, reconfigure the IKE peer to use digital signature authentication.

See the information about Policy Agent and policy applications in *z/OS Communications Server: IP Configuration Reference* for more information about configuring policy.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

IKEv2IKESAKEP.cpp, config_adapter.cpp

**Routing code**

11
**Descriptor code**
7

**Automation**
This message is output to syslog.

**Example**

```
EZD1798I local policy requires the SharedKey parameter but none is specified in KeyExchangeRule 0~3
```

```
EZD1799I IKE cannot initiate with local data addresses ipaddress_range for a Security Association traversing a NAT
```

**Explanation**
The Internet Key Exchange (IKE) daemon tried to activate a phase 2 Security Association (SA) that will traverse a network address translation (NAT) device, but the identity specified for the local data endpoint in the policy for this SA defined a range of local IP addresses that is to be protected by the SA. When traversing a NAT, the IP address of the local data endpoint must be specified as a single host address.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted Security Association (SA). The message instance number precedes the message number in the log output and is used to group related messages from the IKE daemon.

In the message text:
- **ipaddress_range**: The IP address range that was specified for the local data endpoint.

**System action**
The phase 2 SA negotiation fails; IKE daemon processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Ensure that only single host addresses are specified as data endpoints when traversing a NAT. Notify the administrator of the remote security endpoint and ask the administrator to ensure that only single IPv4 addresses are specified as data endpoints when traversing a NAT.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
oakley_phaseII.cpp
Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1799I IKE cannot initiate with local data addresses 9.42.130.0/24 for a Security Association traversing a NAT
EZD1799I IKE cannot initiate with local data addresses 9.42.130.0-9.42.130.128 for a Security Association traversing a NAT

EZD1800I Remote security endpoint at remote_ip port remote_port is using a digital signature for authentication but did not send its certificate in a certificate payload

Explanation
The authentication method that is identified in the authentication payload that was sent by the remote security endpoint uses a digital signature. The Internet Key Exchange (IKE) daemon cannot process that digital signature because the remote security endpoint did not send a certificate payload that contains the certificate that was used to create the digital signature.

In the message text:

remote_ip
  The remote security endpoint IP specification.

remote_port
  The port of the remote security endpoint.

System action
The IKE SA negotiation fails; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that it failed to send a certificate payload.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon
Module
IKEv2IKEAuthRequest.cpp, IKEv2IKEAuthResponse.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1800I Remote security endpoint at 1.2.3.4 port 500 is using a digital signature for authentication, but did not send its certificate in a certificate payload

EZD1801I  jobname STARTING

Explanation
The Communications Server SMTP (CSSMTP) application is starting its initialization.
In the message text:

jobname
The job name of the task that is starting the CSSMTP application.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
None.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlmn
Routing code
10

Descriptor code
12

Automation
This message is written to the system console and syslog. This message is a good candidate for automation. Automation can alert you to when CSSMTP has started.

Example

<table>
<thead>
<tr>
<th>EZD1801I CSSMTP1 STARTING</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1802I jobname INITIALIZATION COMPLETE FOR extWrtName</td>
</tr>
</tbody>
</table>

Explanation
The Communications Server SMTP (CSSMTP) application completed its initialization and is ready to start processing mail for this external writer.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

extWrtName
The external writer name configured on the ExtWrtName statement. See the information about the ExtWrtName statement in z/OS Communications Server: IP Configuration Reference.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlmn
Routing code
10

Descriptor code
12

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP has completed initialization.

Example

<table>
<thead>
<tr>
<th>EZD1802I CSSMTP1 INITIALIZATION COMPLETE FOR CSSMTPEW</th>
</tr>
</thead>
</table>

EZD1803I  jobname SHUTDOWN IN PROGRESS

Explanation
The Communications Server SMTP (CSSMTP) application is shutting down.
In the message text:

*jobname*  
The job name of the task that started the CSSMTP application.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlmn

Routing code
10
Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP is shutting down.

Example

<table>
<thead>
<tr>
<th>EZD18031</th>
<th>CSSMTP1 SHUTDOWN IN PROGRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1804I</td>
<td>jobname SHUTDOWN COMPLETE</td>
</tr>
</tbody>
</table>

Explanation
The Communications Server SMTP (CSSMTP) application ended.
In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
The application ends.

Operator response
Restart CSSMTP if desired.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlmn

Routing code
10

Descriptor code
12
Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP has completed shutting down.

Example

EZD1804I CSSMTP1 SHUTDOWN COMPLETE

EZD1805I  jobname EXITING ABNORMALLY

Explanation
The Communications Server SMTP (CSSMTP) application ended in response to an unexpected problem.

In the message text:

*jobname*

The job name of the task that started the CSSMTP application.

System action
The application ends.

Operator response
Contact the system programmer. If the system programmer indicates that more information is required in the CSSMTP log file, restart CSSMTP with a minimum of LogLevel 79 configured in the configuration file.

System programmer response
Examine the log file to determine the cause of the problem. Take the necessary corrective action and restart CSSMTP. If you need more information to diagnose the errors, restart CSSMTP with a minimum of LogLevel 79. See the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlmn ezamlerr

Routing code
10

Descriptor code
12
Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP shuts down abnormally.

Example

```
EZD1805I CSSMTP1 EXITING ABNORMALLY
```

```
EZD1806I  jobname MODIFY COMMAND UNSUCCESSFUL : reason
```

Explanation
The Communications Server SMTP (CSSMTP) application MODIFY command did not successfully complete.

In the message text:

**jobname**
The job name of the task that started the CSSMTP application.

**reason**
The reason that the MODIFY command failed. Possible values are:

- **CONFIGURATION NOT INITIALIZED**
  The CSSMTP application was unable to process this MODIFY command because message EZD1824E was issued and the CSSMTP application is waiting for a TCP/IP stack to start.

- **INSUFFICIENT STORAGE**
  There is not enough storage to process this modify request.

- **UNKNOWN VERB**
  The command verb is not known. For example: MODIFY procmame,BADVERB

- **MISSING VERB**
  A required command verb is missing. For example: MODIFY procmame,

- **UNKNOWN KEYWORD**
  An unknown or unexpected keyword was specified. For example:
  - BADKEY is the unknown keyword, because MODIFY DISPLAY requires an additional keyword, but BADKEY is not a valid keyword for DISPLAY. Example: MODIFY procmame DISPLAY,BADKEY
  - BADKEY2 is an unexpected keyword, because MODIFY REFRESH does not have additional keywords, but BADKEY2 is an unexpected keyword. Example: MODIFY procmame, REFRESH,BADKEY2

- **MISSING KEYWORD**
  A required keyword is missing. For example: MODIFY procmame,DISPLAY,

- **MISSING VALUE**
  A required value is missing. For example: MODIFY procmame,LOGLEVEL,LEVEL=

- **INCORRECT VALUE**
  An incorrect value was specified. For example: MODIFY procmame,LOGLEVEL,LEVEL=9999

- **PREVIOUS MODIFY COMMAND IN PROGRESS**
  A previous MODIFY REFRESH or MODIFY REFRESHIPLIST command has not completed.

- **COMMAND OUT OF SEQUENCE**
  The SUSPEND command and RESUME command were not issued in correct order. The only exception to the sequencing restriction is that a MODIFY SUSPEND,IMMEDIATE command can be issued after a MODIFY SUSPEND,DELAY command.

- **EXTENDED RETRY NOT ACTIVE**
  The FLUSHRETRY command with the AGE keyword was issued and the extended retry function is not active.

System action
The CSSMTP MODIFY command is ignored.

1190  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Operator response
The operator response is based on the reason value:

**CONFIGURATION NOT INITIALIZED**
Verify whether message EZD1824E is still active. Try the command again after you receive message EZD1840I or EZD1841I.

**INSUFFICIENT STORAGE**
Inform the system programmer of the storage shortage and try the command again after the storage problem has been relieved.

**PREVIOUS MODIFY COMMAND IN PROGRESS**
Wait until the previous MODIFY REFRESH or MODIFY REFRESHIPLIST command has completed before issuing the command again. If the problem persists, then contact the system programmer to determine whether this is a domain name server problem.

All other reason values
Verify that the syntax of the MODIFY command is correct and reissue the command. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator's Commands for the syntax of the command.

If the problem persists, contact the system programmer.

System programmer response
The system programmer response is based on the reason value:

**INSUFFICIENT STORAGE**
See the information about diagnosing storage abends and storage growth in z/OS Communications Server: IP Diagnosis Guide for more information about storage problems.

**PREVIOUS MODIFY COMMAND IN PROGRESS**
Examine the log file to determine the cause of the problem. Take the necessary corrective action.

If you need more information to diagnose the errors, restart the CSSMTP application with a minimum of LogLevel 79. See the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcfg ezamlcfgm

Routing code
Not applicable.

Descriptor code
Not applicable.
Automation
Not applicable.

Example
If the command issued is `f cssmtp1,loglevel,level=999`

<table>
<thead>
<tr>
<th>EZD1806I</th>
<th>CSSMTP1 MODIFY COMMAND UNSUCCESSFUL : INCORRECT VALUE</th>
</tr>
</thead>
</table>

**EZD1807I**  `jobname` ERROR IN INITIALIZATION : `reason`

Explanation
The Communications Server SMTP (CSSMTP) application could not initialize because it detected an error.

In the message text:

**jobname**
The job name of the task that started the CSSMTP application.

**reason**
The reason the initialization failed:

- **CANNOT ACCESS JESSPOOL**
  The CSSMTP application is not defined for access to SAF JESSPOOL resources.

- **CANNOT ACCESS SERVAUTH**
  The CSSMTP application is not defined for access to SAF SERVAUTH resources.

- **CONFIGURATION FILE ERRORS**
  The CSSMTP application found an error in the configuration file.

- **CONFIGURATION FILE NOT AVAILABLE**
  The CSSMTP application cannot open the configuration file.

- **CSSMTP JOBNAME ALREADY STARTED**
  Only one copy of the CSSMTP application can be started with this job name and one copy is already started.

- **EXTERNAL WRITER NAME extWrtName ALREADY IN USE**
  - Only one copy of the CSSMTP application can be started with this external writer name.
  - `extWrtName` is the external writer name.

- **INTERNAL ERRORS**
  Indicates an internal error in the application.

- **INVALID LOG FILE TYPE**
  The log file format PDS or PDSE is not allowed.

- **JES NOT AVAILABLE**
  The CSSMTP application cannot use the functions of JES2 or JES3 subsystems.

- **LOGFILE NOT AVAILABLE**
  The CSSMTP application cannot open the log file.

- **NOT STARTED AS STARTED PROCEDURE**
  The CSSMTP application can be started only as a started procedure.

- **START OPTION ERRORS**
  The CSSMTP application found an error in a start options parameter in the started procedure.

System action
The application ends.
Operator response

The operator response is based on the reason value:

**CSSMTP JOBNAME ALREADY STARTED**

Start this CSSMTP application with a different job name. For example, you could use this form of the start command:

```
s cssmtp.newJobName
```

All other reason values

Contact the system programmer.

System programmer response

The system programmer response is based on the reason value:

**CANNOT ACCESS JESSPOOL**

Define CSSMTP with ALTER access to the local-node.** JESSPOOL class resource. See the information about optional security for CSSMTP in z/OS Communications Server: IP Configuration Guide for details about defining the JESSPOOL class profile for the CSSMTP application.

**CANNOT ACCESS SERVAUTH**

Define CSSMTP with READ access to the EZB.CSSMTP.** SERVAUTH class resource. See the information about optional security for CSSMTP in z/OS Communications Server: IP Configuration Guide for details about defining SERVAUTH class profile for the CSSMTP application.

**EXTERNAL WRITER NAME extWrtName ALREADY IN USE**

Start this CSSMTP application with a different external writer name. See the information about the ExtWrtName statement in z/OS Communications Server: IP Configuration Reference for information about configuring the CSSMTP application.

**CONFIGURATION FILE NOT AVAILABLE**

Ensure that the correct configuration file is configured in the CONFIG DD statement. See the information about the CSSMTP sample started procedure in z/OS Communications Server: IP Configuration Reference for information about configuring CONFIG DD for the CSSMTP application.

**JES NOT AVAILABLE**

Ensure that the JES subsystem is configured correctly. See the information about configuring and starting CSSMTP in z/OS Communications Server: IP Configuration Guide for information about configuring JES subsystems.

**CONFIGURATION FILE ERRORS**

Examine the log file for the configuration error and correct it. See the information about the Communications Server SMTP application in z/OS Communications Server: IP Configuration Reference for information about configuring the CSSMTP application.

**START OPTION ERRORS**

Examine STDOUT and correct the start options parameter that is incorrect. See the information about starting the CSSMTP application in z/OS Communications Server: IP Configuration Reference for information about the start options for the CSSMTP application.

**LOGFILE NOT AVAILABLE**

Ensure that the correct log file is configured in the LOGFILE DD statement. See the information about the CSSMTP sample started procedure in z/OS Communications Server: IP Configuration Reference for information about configuring LOGFILE DD for the CSSMTP application.

**INVALID LOG FILE TYPE**

On the LOGFILE DD statement, define a log file type that is not PDS or PDSE. See the information about the CSSMTP sample started procedure in z/OS Communications Server: IP Configuration Reference for information about configuring LOGFILE DD for the CSSMTP application.

**INTERNAL ERRORS**

Examine the log file for the internal error and correct the error if possible.
NOT STARTED AS STARTED PROCEDURE
Start the CSSMTP application as a started procedure. See the information about configuring and starting CSSMTP in z/OS Communications Server: IP Configuration Guide for details about starting the CSSMTP application.

See the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlmn

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you if the CSSMTP application initialization fails.

Example

EZD1807I CSSMTP1 ERROR IN INITIALIZATION : CONFIGURATION FILE NOT AVAILABLE

EZD1808I jobname MODIFY USEREXIT COMMAND COMPLETED : result

Explanation
The Communications Server SMTP (CSSMTP) application completed processing the MODIFY USEREXIT command.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

result
The result from the MODIFY USEREXIT command. Possible values are:

UPDATED
CSSMTP UserExit was updated based on the modify command level parameter.

NO CHANGES
The value of the level parameter on the CSSMTP MODIFY USEREXIT command is the same as the value currently in use.
The Communications Server SMTP (CSSMTP) application completed processing the MODIFY LOGLEVEL command. In the message text:

**jobname**

The job name of the task that started the CSSMTP application.

**result**

The result from the MODIFY LOGLEVEL command. Possible values are:

**UPDATED**

CSSMTP log level was updated based on the MODIFY command level parameter.
NO CHANGES
The value of the level parameter on CSSMTP MODIFY LOGLEVEL is the same as the value currently in use.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcfm

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
```plaintext
f cssmtp1,loglevel,level=7
EZD1809I CSSMTP1 MODIFY LOGLEVEL COMMAND COMPLETED : UPDATED
```

Explanation
The Communications Server SMTP (CSSMTP) application processed and completed the MODIFY FLUSHRETRY command. Completion means that mail messages currently on the long retry queue for this JES task identifier (TKID) have been moved to the active queue. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information about the FLUSHRETRY command.

In the message text:
jobname
The job name of the task that started the CSSMTP application.

tkid
The task ID that was used in the MODIFY FLUSHRETRY command. The task ID value 0 means that all JES tasks were used.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamllrt

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

f cssmtp1,flushretry,tkid=2
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1810I CSSMTP1 MODIFY FLUSHRETRY,TKID=2 COMMAND COMPLETED

EZD1811I jobname CANNOT WRITE TO LOG FILE

Explaination
The Communications Server SMTP (CSSMTP) application cannot write to the log file configured on the LOGFILE DD in the started procedure. Some common reasons for the failure are that the log file was not large enough or there are system storage problems. This message might be generated when the log level is changed and the error condition still exists. CSSMTP changes the log level dynamically when it starts and stops, so this message can appear multiple times.
In the message text:

**jobname**
The job name of the task that started the CSSMTP application.

**System action**
The CSSMTP application continues processing without logging.

**Operator response**
Contact the system programmer.

**System programmer response**
See the information about the CSSMTP sample started procedure in z/OS Communications Server: IP Configuration Reference for information about configuring LOGFILE DD for the CSSMTP application.

If the storage issue or log file problem is resolved, then issue the MODIFY LOGLEVEL command or the MODIFY REFRESH command to try and restart logging.

**User response**
Not applicable.

**Problem determination**
Contact the system programmer.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamllog

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP has stopped logging.

**Example**

```
EZD1811I CSSMTP1 CANNOT WRITE TO LOG FILE
EZD1813I jobname jesjobname / jesjobid ON wrtname COMPLETED WITH ERRORS. SPOOL FILE DISPOSITION IS HOLD
```
**Explanation**
The Communications Server SMTP (CSSMTP) application completed processing a JES spool file for the job. The disposition of the spool file was changed to HOLD. An error report might be generated, based on the REPORT configuration statement in the configuration file.

In the message text:

*jobname*
  The job name of the task that started the CSSMTP application.

*jesjobname*
  The job name of the JES spool file.

*jesjobid*
  The JES job ID of the spool file.

*wrtname*
  The writer name of the JES spool file.

**System action**
The sysout file is placed in hold status.

**Operator response**
Contact the system programmer.

**System programmer response**
The mail administrator should inspect the notification or log for error messages. If you need more information to diagnose the error, issue the MODIFY LOGLEVEL,LEVEL=31 command to change the log level value.

**User response**
See the system programmer response.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamljes

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.
EZD1813I  CSSMTP USER41P / JOB00115 ON XYZ COMPLETED WITH ERRORS. SPOOL FILE DISPOSITION IS HOLD

EZD1814I  jobname MODIFY RESUME COMMAND COMPLETED

Explanation
The Communications Server SMTP (CSSMTP) application processed and completed the MODIFY RESUME command. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information about the RESUME command.

In the message text:

jobname  The job name of the task that started the CSSMTP application.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamljes

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

f cssmtp1, resume
EZD1834I  CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1814I  CSSMTP1 MODIFY RESUME COMMAND COMPLETED
**EZD1815E  jobname NO ADDRESSES WERE RESOLVED FROM CONFIGURED TARGET SERVERS**

**Explanation**
The Communications Server SMTP (CSSMTP) application detected that no IP addresses were resolved from all TargetName or TargetMx statements in the configuration file.

In the message text:

*jobname*

The job name of the task that started the CSSMTP application.

**System action**
The CSSMTP application will periodically try to resolve addresses again from configured target servers. The application suspends processing until at least one IP address is available. The message will clear when at least one IP address can be resolved.

**Operator response**
Contact the system programmer.

**System programmer response**
Determine if the resolver is initialized.

- If the resolver is not initialized, start the resolver.
- If the resolver is initialized, ensure that the names specified on the TargetServer statement are correct.
  - If the specified names are correct, then there might be a problem in the domain name server setup or the resolver setup. See the information about diagnosing resolver problems in *z/OS Communications Server: IP Diagnosis Guide*.

  After the problem is corrected by fixing the domain name server setup or the resolver setup, then issue the resolver MODIFY REFRESH command to cause the next resolver request to use the updated domain name server setup or resolver setup, and reissue the MODIFY REFRESHIPLIST command to resolve the target servers again.

  - If the specified names are not correct, then correct them and reissue the MODIFY REFRESH command.

If problems persist, then see the information about gathering diagnostic information about CSSMTP problems in *z/OS Communications Server: IP Diagnosis Guide* to determine what documentation you should obtain before contacting IBM Service.

**User response**
Not applicable.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamlres

**Routing code**
1,8
Descriptor code
2,7

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP cannot get any IP address for all configured target servers. You can also use the related message for automation.

```
EZD1802I jobname INITIALIZATION COMPLETE FOR extWrtName
```

Example

```
EZD1815E CSSMTP1 NO ADDRESSES WERE RESOLVED FROM CONFIGURED TARGET SERVERS
EZD1816I jobname jesjobname / jesjobid ON wrtname COMPLETED WITH ERRORS. SPOOL FILE DISPOSITION IS DELETE
```

Explanation
The Communications Server SMTP (CSSMTP) application completed processing a JES spool file for the job. The disposition of the spool file was changed to DELETE. An error report might be generated, based on the REPORT configuration statement in the configuration file.

In the message text:

**jobname**
The job name of the task that started the CSSMTP application.

**jesjobname**
The job name of the JES spool file.

**jesjobid**
The JES job ID of the spool file.

**wrtname**
The writer name of the JES spool file.

**System action**
The sysout spool file is deleted

**Operator response**
Contact the system programmer.

**System programmer response**
The mail administrator should inspect the notification or log file for error messages. If you need more information to diagnose the error, issue MODIFY LOGLEVEL,LEVEL=31 to change the log level value.

**User response**
See the system programmer response.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP: CSSMTP
Module
ezmailjes

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1816I CSSMTP USER41P / JOB00116 ON XYZ COMPLETED WITH ERRORS. SPOOL FILE DISPOSITION IS DELETE

EZD1817I  jobname UNABLE TO CONNECT TO TARGET SERVER ipaddress

Explanation
The Communications Server SMTP (CSSMTP) application cannot connect to the specified target server IP address.

In the message text:

jobname
  The job name of the task that started the CSSMTP application.

ipaddress
  The IP address of the target server.

System action
The CSSMTP application attempts to connect to another configured target server IP address. CSSMTP will periodically try to connect to this target server IP address based on the configured TIMEOUT ConnectRetry value.

If the CSSMTP application connects to the target server, then message EZD1821I is issued to display the IP address of the target server.

Operator response
Contact the system programmer.

System programmer response
If the IP address that is displayed in this message is not correct, then issue the MODIFY DISPLAY,IPLIST command to determine whether the IP address was specified in the configuration file or was the result of a host name or MX name resolution. If you change the IP address, host name, or MX name in the configuration file, then issue a MODIFY REFRESH command to update the configuration settings. If the IP address is incorrect and is the result of a name resolution, then diagnose the resolver or Domain Name System (DNS) problems. If the IP address displayed in this message is correct, then check for network connectivity problems.

Examine the log file to determine the cause of the problem. Take the necessary corrective action based on the log information that indicates the cause of the failure. If you need more information to diagnose the error, issue MODIFY LOGLEVEL,LEVEL=47 to change log level value. Check the application trace log for failures. Send a ping to the target server IP address from the system. If the ping works, check the configured port for this target server and ensure that the remote SMTP server is running and listening on that port.
See the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

**User response**
Not applicable.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamlcon

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to the need to take corrective action when CSSMTP is not able to send any mail to this IP address.

**Example**

```
EZD1817I CSSMTP1 UNABLE TO CONNECT TO TARGET SERVER 9.1.1.1
```

```
EZD1818I  jobname UNABLE TO SEND TO TARGET SERVER ipaddress
```

**Explanation**
The Communications Server SMTP (CSSMTP) application cannot send mail to the target server IP address. Errors occurred after the connection to the target server was successful but before a successful EHLO or HELO SMTP command response was received.

In the message text:

- **jobname**
  The job name of the task that started the CSSMTP application.

- **ipaddress**
  The IP address of the target server.

**System action**
The CSSMTP application attempts to send mail to another target server IP address, if one is available. CSSMTP periodically tries to establish communication with this target server IP address, based on the configuration statement TIMEOUT ConnectRetry value, because the connection must be re-established. If the CSSMTP application connects to the target server, then message EZD1821I is issued to display the IP address of the target server.
Operator response
Contact the system programmer.

System programmer response
If the IP address that is displayed in this message is not correct, then issue the MODIFY DISPLAY,IPLIST command to determine whether the IP address was specified in the configuration file or was the result of a host name or MX name resolution. If you change the IP address, host name, or MX name in the configuration file, then issue a MODIFY REFRESH command to update the configuration settings. If the IP address is incorrect and is the result of a name resolution, then diagnose the resolver or Domain Name System (DNS) problems. If the IP address displayed in this message is correct, then check for network connectivity problems.

Examine the log file to determine the cause of the problem. Take the necessary corrective action. If you need more information to diagnose the error, issue MODIFY LOGLEVEL,LEVEL=47 to change log level value. Check the application trace log for failures. Try pinging the target server IP address from the system. If the ping works, check the configured port for this target server and ensure that the remote SMTP server is running and listening on that port. For example, if the server is running but does not send the protocol greeting after a successful connect() call, the server that is using the port might not be an SMTP server.

See the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcon

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to the need to take corrective action when CSSMTP is not able to send any mail to this IP address.

Example

EZZ1818I CSSMTP1 UNABLE TO SEND TO TARGET SERVER 9.1.1.1

EZD1819I  jobname UNABLE TO ESTABLISH A TLS CONNECTION TO TARGET SERVER ipaddress
**Explanation**
The Communications Server SMTP (CSSMTP) application was unable to establish a TLS connection to the target server IP address. Possible causes might be that TLS is not supported on the target server, that security certificates need updating, or that the Policy Agent is not started.

In the message text:

- **jobname**
  - The job name of the task that started the CSSMTP application.

- **ipaddress**
  - The IP address of the target server.

**System action**
The CSSMTP application attempts to find another configured target server IP address that supports TLS. CSSMTP periodically tries to establish a TLS connection to this target server IP address based on the configuration statement TIMEOUT ConnectRetry value.

**Operator response**
Contact the system programmer.

**System programmer response**
If the IP address displayed in this message does not require security, then change the configuration TARGETSERVER statement SECURE parameter value to NO in the configuration file and issue a MODIFY REFRESH command to update the configuration settings. If the IP address displayed in this message does require security, then check for other problems such as a server capability that needs updating, security certificates that need updating, or a Policy Agent that is not started.

Examine the log file to determine the cause of the problem. Take the necessary corrective action based on the log information that indicates the cause of the failure. If you need more information to diagnose the error, issue MODIFY LOGLEVEL,LEVEL=47 to change log level value. Check the application's trace log for failures.

See the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

**User response**
Not applicable.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamlcon

**Routing code**
10

**Descriptor code**
12
### Automation

This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to the need to take corrective action when CSSMTP is not able to send any mail to this IP address.

### Example

<table>
<thead>
<tr>
<th>EZD1819I</th>
<th>CSSMTP1 UNABLE TO ESTABLISH A TLS CONNECTION TO TARGET SERVER 9.1.1.1</th>
</tr>
</thead>
</table>

**EZD1820E**  
**jobname** NO TARGET SERVER IS CAPABLE OF RECEIVING MAIL AT THIS TIME

### Explanation

The Communications Server SMTP (CSSMTP) application detected that there is no configured target server that is capable of receiving messages at this time.

In the message text:

**jobname**  
The job name of the task that started the CSSMTP application.

### System action

The CSSMTP application is unable to get any target server to take mail at this time. CSSMTP periodically tries to communicate with the configured target servers. This message will clear when CSSMTP can communicate with at least one target server IP address.

### Operator response

Contact the system programmer.

### System programmer response

Check the system console log first for message EZD1824E to determine whether the application can communicate with a TCP/IP stack. If stack communication is not the problem, continue to check the system console log to determine whether message EZD1817I, EZD1818I, or EZD1819I was issued before this message. There might be more than one of these messages, depending on the number of server target IP addresses that are currently being used by the application.

Examine the log file to determine the cause of the problem. Take the necessary corrective action. If you need more information to diagnose the error, issue MODIFY LOGLEVEL,LEVEL=47 to change log level value. Check the application trace log for failures.

See the information about gathering diagnostic information about CSSMTP problems in *z/OS Communications Server: IP Diagnosis Guide* to determine what documentation you should obtain before contacting IBM Service.

### User response

Not applicable.

### Problem determination

See the system programmer response.

### Source

z/OS Communications Server TCP/IP: CSSMTP

### Module

ezamlsmt
Routing code
1,8

Descriptor code
2,7

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to the need to take corrective action as soon as possible because CSSMTP is not able to send any mail at this time.

Example

```
EZD1820E CSSMTP1 NO TARGET SERVER IS CAPABLE OF RECEIVING MAIL AT THIS TIME
```

```
EZD1821I  jobname ABLE TO USE TARGET SERVER ipaddress
```

Explanation
The Communications Server SMTP (CSSMTP) application detected that the target server IP address is available for receiving mail.

In the message text:

- **jobname**
  The job name of the task that started the CSSMTP application.

- **ipaddress**
  The IP address of the target server.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlsmt
Routing code
10

Descriptor code
12

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when a target server is available.

Example

EZD1821I CSSMTP1 ABLE TO USE TARGET SERVER 9.1.1.1

EZD1822I  jobname MODIFY SUSPEND type COMMAND COMPLETED

Explanation
The Communications Server SMTP (CSSMTP) application processed and completed the MODIFY SUSPEND command. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information about the SUSPEND command and the different type values.

In the message text:

**jobname**
The job name of the task that started the CSSMTP application.

**type**
The type of SUSPEND command that completed.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamljes
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

f cssmtp1,suspend,immediate
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1822I CSSMTP1 MODIFY SUSPEND IMMEDIATE COMMAND COMPLETED

EZD1823I  jobname MODIFY FLUSHRETRY,AGE=age COMMAND COMPLETED

Explanation

The Communications Server SMTP (CSSMTP) application processed and completed the MODIFY FLUSHRETRY command. Mail messages that were on the extended retry queue and that were older than the AGE value specified on the command were moved to the active queue. See the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator’s Commands for information about the FLUSHRETRY command.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

age
The age value that was specified on the MODIFY FLUSHRETRY command. The age value 0 means that all messages were moved.

System action
Processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP
Module
ezamlert

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
f cssmtp1,flushretry,age=1
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1823I CSSMTP1 MODIFY FLUSHRETRY,AGE=1 COMMAND COMPLETED

EZD1824E  jobname WAITING FOR A TCPIP STACK

Explanation
The Communication Server SMTP (CSSMTP) application cannot communicate with a TCP/IP stack. This message
might be issued during CSSMTP initialization or during mail processing.

There are four possible causes of this error:

-p tcpStack
The start option stack is not available.

_BPXK_SETIBMOPT_TRANSPORT tcpStack
The environment variable stack is not available.

No TCP/IP affinity
No stack is available.

EZB.STACKACCESS
Stack security settings prevent communication with a stack.

See the information about starting the CSSMTP application in z/OS Communications Server: IP Configuration
Reference for information about stack affinity.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
CSSMTP initialization or mail processing does not complete and CSSMTP periodically tries to communicate with
a stack. This message will clear after CSSMTP connects to the TCP/IP stack.

Operator response
Save the log file and contact the system programmer.

System programmer response
Examine the log file to help determine the cause of the problem and which stack needs to be started. Based on
the following, take the necessary corrective action:

- If you are using the -p start option, then verify that the specified stack is started.
• If you are using the _BPXK_SETIBMOPT_TRANSPORT environment variable, then verify that the stack configured on the environment variable is started.
• If you are not using TCP/IP affinity, then verify that a stack is available.
• If you are restricting stack access by using the SERVAUTH class resource EZB.STACKACCESS then verify that the CSSMTP application has authority to access to the resource.

If you need more information during mail processing time to diagnose the warnings, issue MODIFY LOGLEVEL,LEVEL=79 to increase the log level. Check the application trace log for failures when performing the TCP/IP socket() function. The inability to obtain a socket blocks communication between the application and the TCP/IP stack.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcfg

Routing code
1,8

Descriptor code
2,7

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP cannot communicate with a stack.

Example

```
EZD1824E CSSMTP1 WAITING FOR A TCPIP STACK
```

```
EZD1825I jobname JES NOT AVAILABLE
```

Explanation
The Communications Server SMTP (CSSMTP) application is ending because JES2 or JES3 subsystems are no longer available.

In the message text:

`jobname`
The job name of the task that started the CSSMTP application.

System action
The CSSMTP application ends.
Operator response
If the JES2 or JES3 subsystem was not stopped, contact the system programmer. If the system programmer indicates that more information is required in the CSSMTP log file, restart CSSMTP with the minimum value LOGLEVEL,LEVEL=95 configured in the main configuration file. If the JES2 or JES3 subsystems were stopped, restart them before restarting CSSMTP.

System programmer response
Examine the log file to determine the cause of the problem. Take the necessary corrective action and restart CSSMTP. If you need more information to diagnose the errors, restart CSSMTP with the minimum value LOGLEVEL,LEVEL=95. If the log shows SAPI rc=0 SSOBRETN=32 SSS2REAS=36 and the JES2 DESTDEF statement specifies NODENAME=REQUIRED, you must define the writer name specified in the ExtWrtName statement to JES2. You can dynamically define it to JES using the command

$ADD DESTID(xxxxxxxx),DEST=xxxxxxxxx,

where xxxxxxxx is the wtrName value specified in the ExtWrtName statement. See z/OS JES2 Initialization and Tuning Reference for information about the DESTDEF statement. To permanently add the destination, use the JES initialization DEStid statement: DESTID(xxxxxxxx) DEST=xxxxxxxxx. See DEStid(xxxxxxxx) - Route Code Name in z/OS JES2 Initialization and Tuning Reference for information about the DEStid JES2 statement. See the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlmn

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP shuts down abnormally.

Example
Ezd1825i  cssmtp1  jes  not  available

Ezd1826i  jobname  cannot  write  to  z/os  unix  file  system  dead  letter  directory
**Explanation**
The Communications Server SMTP (CSSMTP) application cannot write to the configured or default z/OS UNIX file system dead letter directory.

In the message text:

*jobname*

The job name of the task that started the CSSMTP application.

**System action**
The CSSMTP application continues processing without storing dead letters and the DeadLetterAction configuration statement is set to Delete.

**Operator response**
Contact the system programmer.

**System programmer response**
Possible reasons for this message are:

- The UNIX file system dead letter directory is full or nearly full.
- The UNIX file system dead letter directory is not accessible in write mode.

If you determine that the dead letter directory is full or nearly full, then do one of the following:

- Perform the following steps to free some space in the directory:
  1. Issue the MODIFY SUSPEND command to suspend new spool file processing.
  2. Delete all unneeded dead letters.
  3. Issue the MODIFY REFRESH command to enable the application to write dead letters into the configured or default directory again.
  4. Issue the MODIFY RESUME command to resume new spool file processing.

- Perform the following steps to use a larger directory:
  1. Allocate a larger z/OS UNIX file system for the dead letter directory.
  2. Change the DeadLetterDirectory configuration statement to specify the new directory.
  3. Issue a MODIFY REFRESH command to use the larger directory.

If you determine that the UNIX file system dead letter directory is not accessible in write mode, perform the following steps:

1. Do one of the following:
   - Mount the UNIX file system in write mode.
   - Specify a directory to which you have write access in the DeadLetterDirectory configuration statement in the CSSMTP configuration file.

2. Issue a MODIFY REFRESH command to use the new directory.

See the information about the UNDELIVERABLE statement in z/OS Communications Server: IP Configuration Reference for more information about z/OS UNIX file system dead letter directory.

**User response**
Not applicable.

**Problem determination**
See the system programmer response.
Automation

This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP has stopped storing dead letters into the z/OS UNIX file system dead letter directory.

Example

```
EZD1826I CSSMTP1 CANNOT WRITE TO z/OS UNIX FILE SYSTEM DEAD LETTER DIRECTORY
EZD1827I jobname CANNOT WRITE TO THE z/OS UNIX FILE SYSTEM MAIL DIRECTORY RETURN CODE errno REASON CODE errnojr
```

Explanation

The Communications Server SMTP (CSSMTP) application cannot write to the configured or default z/OS UNIX file system mail directory associated with the EXTENDEDRETRY statement.

In the message text:

- **jobname**
  The job name of the task that started the CSSMTP application.

- **errno**
  The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) in z/OS UNIX System Services Messages and Codes.

- **errnojr**
  The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the return codes (ernos) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action

The CSSMTP application stops.

Operator response

Contact the system programmer.

System programmer response

Check the CSSMTP log for the errno/errnojr associated with the I/O error on the MailDirectory parameter. Look up the return code and reason code values to determine the next action to take. Take corrective action and restart CSSMTP.
If you determine that the z/OS UNIX file system mail directory is not accessible in write mode, perform one of the following steps:

- Mount the z/OS UNIX file system in write mode. Restart the CSSMTP application.
- Specify a new directory on the MailDirectory parameter of the ExternalRetry configuration statement in the CSSMTP configuration file. This directory must be one to which CSSMTP will have write access.

See the ExtendedRetry statement in z/OS Communications Server: IP Configuration Reference for more information about the z/OS UNIX file system mail directory.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: CSSMTP

**Module**

ezamlert

**Routing code**

10

**Descriptor code**

12

**Automation**

This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when CSSMTP has a problem storing mails into the z/OS UNIX file system mail directory.

**Example**

```
EZD1827I CSSMTP1 CANNOT WRITE TO THE z/OS UNIX FILE SYSTEM MAIL DIRECTORY RETURN CODE 111
REASON CODE EF086015
```

```
EZD1828I jobname DISPLAY LOGLEVEL = loglevel
```

**Explanation**

The message is issued in response to a Communications Server SMTP (CSSMTP) application MODIFY jobname,DISPLAY,LOGLEVEL command.

In the message text:

- **jobname**
  - The job name of the task that started the CSSMTP application.
- **loglevel**
  - The current log level value that is in effect. This value is specified with the LogLevel configuration statement or with the MODIFY LOGLEVEL,LEVEL command.
  - The value displayed is the arithmetic sum of the currently active values from the following list.
None. No messages are logged.

Error-level messages are logged.

Warning-level messages are logged.

Event-level messages are logged.

Info-level messages are logged.

Message-level JES (spool) messages are logged. This level traces the CSSMTP commands and command syntax parser replies between the spool and CSSMTP.

Message-level TCP/IP messages are logged. This level traces the CSSMTP commands and remote SMTP server replies between CSSMTP and the TCP/IP network.

Debug-level messages are logged. These are internal debug messages intended for development and IBM service use only.

Trace-level messages are logged. These are function entry and exit traces that show the path through the code. This level is intended for development and IBM service use only.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcfm

Routing code
10
**EZD1829I**  *jobname* CONFIGURATION:

**Explanation**
This message is a header message that is issued in response to a Communications Server SMTP (CSSMTP) application MODIFY DISPLAY,CONFIG command. This message displays the configuration values. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator's Commands for a description of the display.

In the message text:

*jobname*
The job name of the task that started the CSSMTP application.

**System action**
CSSMTP processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamlcffm

**Routing code**
10

**Descriptor code**
12
Automation
Not applicable.

Example

```
F CSSMTP,D,CONFIG
EZD1829I CSSMTP CONFIGURATION:
EZD1830I  jobname IPLIST:
```

Explanation
This message is a header message that is issued in response to a Communications Server SMTP (CSSMTP) application MODIFY DISPLAY,IPLIST command. This message is followed by information about the IP addresses for all the target servers. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator's Commands for an example of the MODIFY DISPLAY,IPLIST command display.

In the message text:

**jobname**
The job name of the task that started the CSSMTP application.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamicfmc

Routing code
10

Descriptor code
12

Automation
Not applicable.
Example

EZD1830I CSSMTP IPLIST:

EZD1831I  jobname  TARGETS:

Explanation
This message is a header message that is issued in response to a Communications Server SMTP (CSSMTP) application MODIFY DISPLAY,TARGETS command. This message is followed by global and specific information about target servers. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator's Commands for an example of the MODIFY DISPLAY,TARGETS command display.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
The CSSMTP application continues processing.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcfm

Routing code
10

Descriptor code
12

Automation
Not applicable.
Example

F  CSSMTP,DISPLAY,TARGETS
EZD1831I  CSSMTP  TARGETS:

EZD1832I  jobname  SPOOLSTATUS:

Explanation

This message is a header message that is issued in response to a Communications Server SMTP (CSSMTP) application MODIFY DISPLAY,SPOOLSTATUS or MODIFY DISPLAY,SPOOLSTATUS,SUMMARY command. This message is followed by the global and specific status of the current or previous JES spool information. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator's Commands for an example of the MODIFY DISPLAY,SPOOLSTATUS and MODIFY DISPLAY,SPOOLSTATUS,SUMMARY command display.

In the message text:

jobname

The job name of the task that started the CSSMTP application.

System action

Processing continues.

Operator response

None.

System programmer response

None.

User response

Not applicable.

Problem determination

None.

Source

z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcfm

Routing code

10

Descriptor code

12

Automation

Not applicable.
Example

**EZD1832I CSSMTP1 SPOOLSTATUS:**

**Explanation**

This message is a header message that is issued in response to a Communications Server SMTP (CSSMTP) application MODIFY DISPLAY, SPOOLSTATUS, DETAIL command. This message is followed by the global and specific status of the current or previous JES spool information. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator’s Commands for an example of the MODIFY DISPLAY, SPOOLSTATUS, DETAIL command display.

In the message text:

*jobname*

  The job name of the task that started the CSSMTP application.

**System action**

CSSMTP processing continues.

**Operator response**

None.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: CSSMTP

**Module**

ezamlcfm

**Routing code**

10

**Descriptor code**

12

**Automation**

Not applicable.
EZD1834I  **jobname** MODIFY COMMAND ACCEPTED

**Explanation**
This message is issued in response to a Communications Server SMTP (CSSMTP) application MODIFY command. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator's Commands for information about the MODIFY command.

In the message text:

**jobname**
The job name of the task that started the CSSMTP application.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamlcfc

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.
EZD1835I  jobname CHKPOINT DD NOT FOUND. CHECKPOINT FUNCTION NOT AVAILABLE

Explanation
The CHKPOINT data definition statement was missing or the data set name was NULLFILE in the Communications Server SMTP (CSSMTP) started procedure.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
The CSSMTP application continues without checkpoint functions.

Operator response
Contact the system programmer.

System programmer response
If you want the checkpoint functions, then define the checkpoint data set and update the CSSMTP started procedure. See the information about the CSSMTP sample started procedure in z/OS Communications Server: IP Configuration Reference for information about configuring CHKPOINT DD for the CSSMTP application.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlckp

Routing code
10

Descriptor code
12

Automation
Not applicable
EZD1835I CSSMTP CHKPOINT DD NOT FOUND. CHECKPOINT FUNCTION NOT AVAILABLE

**EZD1836I**  *jobName* CHECKPOINT FAILED. REASON CODE = *reasonCode*

**Explanation**
There was a failure while the Communications Server SMTP (CSSMTP) application checkpoint function was initializing.

In the message text:

*jobname*
- The job name of the task that started the CSSMTP application.

*reasonCode*
- A code describing the error. Possible values are:
  - 12
    - Locks could not be initialized.
  - 14
    - Storage was not available.
  - 30
    - The checkpoint file was not available.
  - 31
    - The VSAM linear data set could not be connected to the application.

**System action**
The application ends.

**Operator response**
Contact the system programmer.

**System programmer response**
Use the log to determine the exact failure. Ensure that the LogLevel includes ERROR.

**User response**
Not applicable.

**Problem determination**
Check the console log for additional messages, for example message IEC161I. Check the log for Checkpoint Open messages.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamickp

**Routing code**
10
**EZD1837I**  
*jobname* CONFIGURATION UNCHANGED

**Explanation**

The Communications Server SMTP (CSSMTP) application processed the configuration file as the result of a MODIFY REFRESH command. The configuration file definitions match the existing configuration.

In the message text:

*jobname*

The job name of the task that started the CSSMTP application.

**System action**

CSSMTP processing continues.

**Operator response**

None.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: CSSMTP

**Module**

ezamlcfg

**Routing code**

10

**Descriptor code**

12
Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>Command</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>f cssmtp1,refresh</td>
<td>EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED</td>
</tr>
<tr>
<td></td>
<td>EZD1837I CSSMTP1 CONFIGURATION UNCHANGED</td>
</tr>
<tr>
<td></td>
<td>EZD1848I CSSMTP1 MODIFY REFRESH COMMAND COMPLETED</td>
</tr>
</tbody>
</table>

**EZD1838I  jobname CONFIGURATION UNCHANGED WITH WARNINGS**

**Explanation**
The Communications Server SMTP (CSSMTP) application processed the configuration file as the result of a MODIFY REFRESH command. Warnings were issued while the configuration file was being processed. The CSSMTP configuration is unchanged.

In the message text:

*jobname*

The job name of the task that started the CSSMTP application.

**System action**
CSSMTP processing continues.

**Operator response**
Save the log file and contact the system programmer.

**System programmer response**
Examine the log file to determine the cause of the WARNING messages in the log. Take the necessary corrective action. If you need more information to diagnose the warnings, issue MODIFY LOGLEVEL,LEVEL=79 to increase the log level. Issue the MODIFY REFRESH command.

See the information about the Communications Server SMTP application in *z/OS Communications Server: IP Configuration Reference* for information about configuring CSSMTP.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamlcfg

**Routing code**
10
Descriptor code
12

Automation
Not applicable.

Example

f cssmtp1,refresh
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1838I CSSMTP1 CONFIGURATION UNCHANGED WITH WARNINGS
EZD1848I CSSMTP1 MODIFY REFRESH COMMAND COMPLETED

EVD1839I  jobname  CONFIGURATION NOT UPDATED BECAUSE ERRORS WERE FOUND IN CONFIGURATION FILE

Explanation
The Communications Server SMTP (CSSMTP) application processed the configuration file as the result of a MODIFY REFRESH command. The existing configuration was not updated, because errors were issued while the configuration file was being processed.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
CSSMTP processing continues.

Operator response
Save the log file and contact the system programmer.

System programmer response
Examine the log file to determine the cause of the ERROR messages in the log. Take the necessary corrective action. If you need more information to diagnose the warnings, issue MODIFY LOGLEVEL,LEVEL=79 to change the log level value. Issue the MODIFY REFRESH command.

See the information about the Communications Server SMTP application in z/OS Communications Server: IP Configuration Reference for information about configuring CSSMTP.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcfg
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

f cssmtp1,refresh  
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED  
EZD1839I CSSMTP1 CONFIGURATION NOT UPDATED BECAUSE ERRORS WERE FOUND IN CONFIGURATION FILE  
EZD1848I CSSMTP1 MODIFY REFRESH COMMAND COMPLETED

EZD1840I    jobname UPDATED CONFIGURATION

Explanation
The Communications Server SMTP (CSSMTP) application processed the configuration file during initialization of the application or as a result of a MODIFY REFRESH command. The existing configuration was updated.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
CSSMTP processing continued.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcfg

Routing code
10
Explanation
The Communications Server SMTP (CSSMTP) application processed the configuration file during initialization of the application or as a result of a MODIFY REFRESH command. The existing configuration was updated, but warnings were issued while the configuration file was being processed.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
CSSMTP processing continues.

Operator response
Save the log file and contact the system programmer.

System programmer response
Examine the log file to determine the cause of the WARNING messages in the log. Take the necessary corrective action. If you need more information to diagnose the warnings, issue MODIFY LOGLEVEL,LEVEL=79 to change the log level value. Issue MODIFY REFRESH command.

See the information about the Communications Server SMTP application in z/OS Communications Server: IP Configuration Reference for information about configuring CSSMTP.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlcfg
Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
f cssmtp1,refresh
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1841I CSSMTP1 UPDATED CONFIGURATION WITH WARNINGS
EZD1846I CSSMTP1 UPDATED TARGET SERVERS
EZD1848I CSSMTP1 MODIFY REFRESH COMMAND COMPLETED

EZD1842I  jobname MODIFY REFRESHIPLIST COMMAND COMPLETED

Explanation
The Communications Server SMTP (CSSMTP) application processed and completed the MODIFY
REFRESHIPLIST command.
In the message text:

  jobname
    The job name of the task that started the CSSMTP application.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlres

Routing code
10
EZD1843I  jobname TARGET SERVERS UNCHANGED

Explanation
The Communications Server SMTP (CSSMTP) application processed the TargetServer statements in the configuration file as a result of the MODIFY REFRESH command or MODIFY REFRESHIPLIST command. No updated IP addresses were resolved from the TargetServer configuration statements. The current IP addresses will continue to be used.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlres

Routing code
10
Explanation

The Communications Server SMTP (CSSMTP) application processed the TargetServer statements in the configuration file as a result of the MODIFY REFRESH command or MODIFY REFRESHIPLIST command. No updated IP addresses were resolved from the TargetServer statements. The current IP addresses will continue to be used. Warnings were issued as a result of processing the TargetServer statements.

The target name on a TargetServer statement might have resulted in one of the following:

- A target name that could not be resolved to any IP address
- A target name that was resolved to more than 4 IP addresses

In the message text:

**jobname**

The job name of the task that started the CSSMTP application.

System action

CSSMTP processing continues.

Operator response

Save the system log for problem determination and contact the system programmer.

System programmer response

Examine the log file to determine the cause of the WARNING messages in the log. These warnings might not cause the CSSMTP application to be unable to deliver mail, but they might indicate problems.

Ensure that the names specified on the TargetServer statement are correct.

- If the specified names are correct, then there might be a problem in the domain name server setup or the resolver setup. See the information about diagnosing resolver problems in z/OS Communications Server: IP Diagnosis Guide for information about domain name server or resolver problems. After you correct the problem by fixing the domain name server setup or the resolver setup, then issue the resolver MODIFY REFRESH command to cause the next resolver request to use the updated domain name server setup or the resolver setup, and reissue the MODIFY REFRESHIPLIST command to resolve the target servers again.

- If the specified names are not correct, then correct them and reissue the MODIFY REFRESH command.

If problems persist, then see the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.
User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlres

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
f cssmtp1,refreshiplist
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1844I CSSMTP1 TARGET SERVERS UNCHANGED WITH WARNINGS
EZD1842I CSSMTP1 MODIFY REFRESHIPLIST COMMAND COMPLETED

EZD1845I  jobname TARGET SERVERS NOT UPDATED BECAUSE OF RESOLVER ERRORS

Explanation
The Communications Server SMTP (CSSMTP) application processed the TargetServer statements in the configuration file as a result of the MODIFY REFRESH command or MODIFY REFRESHIPLIST command. No IP addresses were resolved from the TargetServer statements. The current IP addresses will continue to be used.

In the message text:

jobname  The job name of the task that started the CSSMTP application.

System action
CSSMTP processing continues.

Operator response
Save the system log for problem determination and contact the system programmer.

System programmer response
Examine the log file to determine the cause of the ERROR messages in the log. These errors might not cause the CSSMTP application to be unable to deliver mail, but they might indicate problems.

Determine whether the resolver is initialized.
• If the resolver is not initialized, start the resolver.
• If the resolver is initialized, ensure that the names specified on the TargetServer statement are correct.

  – If the specified names are correct, then there might be a problem in the domain name server setup or the resolver setup. See the information about diagnosing resolver problems in z/OS Communications Server: IP Diagnosis Guide for information about domain name server or resolver problems. After you correct the problem by fixing the domain name server setup or the resolver setup, then issue the resolver MODIFY REFRESH command to cause the next resolver request to use the updated domain name server setup or resolver setup, and reissue the MODIFY REFRESHIPLIST command to resolve the target servers again.

  – If the specified names are not correct, then correct them and reissue the MODIFY REFRESH command.

See the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlres

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

```
f cssmtp1,refreshiplist
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1845I CSSMTP1 TARGET SERVERS NOT UPDATED BECAUSE OF RESOLVER ERRORS
EZD1842I CSSMTP1 MODIFY REFRESHIPLIST COMMAND COMPLETED
```

**EZD1846I**  
*jobname* UPDATED TARGET SERVERS

Explanation

The Communications Server SMTP (CSSMTP) application processed the TargetServer statements in the configuration file during initialization of the application or as a result of the MODIFY REFRESH command or MODIFY REFRESHIPLIST command. IP addresses and their attributes were updated as a result of the TargetServer statements.

In the message text:

**jobname**

The job name of the task that started the CSSMTP application.
System action
CSSMTP processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlres

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
f cssmtp1,refreshiplist
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1846I CSSMTP1 UPDATED TARGET SERVERS
EZD1842I CSSMTP1 MODIFY REFRESHIPLIST COMMAND COMPLETED

EZD1847I  jobname UPDATED TARGET SERVERS WITH WARNINGS

Explanation
The Communications Server SMTP (CSSMTP) application processed the TargetServer statements during initialization of the application or as a result of the MODIFY REFRESH command or MODIFY REFRESHIPLIST command. IP addresses and their attributes were updated as a result of the TargetServer statements. Warnings were issued as a result of processing TargetServer statements.

The target name on a TargetServer statement might have resulted in one of the following:
• A target name that could not be resolved to any IP address.
• A target name that was resolved to more than 4 IP addresses.
• Configured or resolved IP addresses that exceeded the maximum of 4.
In the message text:

**jobname**

The job name of the task that started the CSSMTP application.

**System action**

CSSMTP processing continues.

**Operator response**

Save the system log for problem determination and contact the system programmer.

**System programmer response**

Examine the log file to determine the cause of the WARNING messages in the log. These warnings might not cause the CSSMTP application to be unable to deliver mails, but they might indicate problems.

Ensure that the names specified on the TargetServer statement are correct.

- If the specified names are correct, then there might be a problem in the domain name server setup or the resolver setup. See the information about diagnosing resolver problems in z/OS Communications Server: IP Diagnosis Guide for information about domain name server or resolver problems. After you correct the problem by fixing the domain name server setup or the resolver setup, then issue the resolver MODIFY REFRESH command to cause the next resolver request to use the updated domain name server setup or resolver setup, and reissue the MODIFY REFRESHIPLIST command to resolve the target servers again.
- If the specified names are not correct, then correct them and reissue the MODIFY REFRESH command.

See the information about gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide to determine what documentation you should obtain before contacting IBM Service.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: CSSMTP

**Module**

ezamlres

**Routing code**

10

**Descriptor code**

12

**Automation**

Not applicable.
Example

`f cssmtp1,refreshiplist`
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED
EZD1847I CSSMTP1 UPDATED TARGET SERVERS WITH WARNINGS
EZD1842I CSSMTP1 MODIFY REFRESHIPLIST COMMAND COMPLETED

**EZD1848I**  `jobname MODIFY REFRESH COMMAND COMPLETED`  

**Explanation**
The Communications Server SMTP (CSSMTP) application processed and completed the MODIFY REFRESH command.

In the message text:

*`jobname`*

The job name of the task that started the CSSMTP application.

**System action**
CSSMTP processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamlres ezamlcfg

**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable.

**Example**

`f cssmtp1,refresh`
EZD1834I CSSMTP1 MODIFY COMMAND ACCEPTED

---

1238  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
EZD1849I  jobname jesjobname / jesjobid ON wrtname RESTART FAILED

Explanation
The Communications Server SMTP (CSSMTP) application attempted to restart the named spool file as a result of a warm start. CSSMTP was previously running with checkpointing enabled. The spool file was unavailable.

In the message text:

jobname
   The job name of the task that started the CSSMTP application.

jesjobname
   The job name of the JES spool file.

jesjobid
   The JES job ID of the spool file.

wrtname
   The writer name of the JES spool file.

System action
Processing continues without the spool file.

Operator response
Contact the system programmer

System programmer response
Determine the status of the spool file. The spool file must have the same writer name or destination that it had when the spool file was first processed by the CSSMTP application.

User response
Not applicable.

Problem determination
Use the System Display and Search Facility (SDSF) to determine the status of the spool file.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamclkp

Routing code
10

Descriptor code
12
Automation
Not applicable

Example

EZD1849I CSSMTP USER41P / JOB000115 ON XYZ RESTART FAILED

EZD1850I  jobname jesjobname / jesjobid ON wrtname RESTARTED

Explanation
The Communications Server SMTP (CSSMTP) application restarted the named spool data set. A spool file can be restarted when the processing of the spool file was not completed by the previous instance of the CSSMTP application.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

jesjobname
The job name of the JES spool file.

jesjobid
The JES job ID of the spool file.

wrtname
The writer name of the JES spool file.

System action
Processing for the spool data set continues.

Operator response
Not applicable

System programmer response
Not applicable

User response
Not applicable.

Problem determination
Not applicable

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlckp

Routing code
10
**Descriptor code**

12

**Automation**

Not applicable

**Example**

| EZD1850I | CSSMTP USER41P / JOB00115 ON XYZ RESTARTED |

**EZD1851I**  
*jobname* *jesjobname* / *jesjobid* ON *wrtname* INPUT ERROR. *errnoText*

**Explanation**

The Communications Server SMTP (CSSMTP) application has encountered an error while it was reading the JES spool file.

In the message text:

*jobname*  
The job name of the task that started the CSSMTP application.

*jesjobname*  
The job name of the JES spool file.

*jesjobid*  
The JES job ID of the spool file.

*wrtname*  
The writer name of the JES spool file.

*errnoText*  
The error message from the file system that describes the error.

**System action**

Reading of the spool file is discontinued.

**Operator response**

Notify the system programmer.

**System programmer response**

Determine the cause of the input error.

**User response**

Not applicable.

**Problem determination**

See the system programmer response.

**Source**

z/OS Communications Server TCP/IP: CSSMTP

**Module**

ezamljes
**Routing code**
10

**Descriptor code**
12

**Automation**
Not applicable

**Example**

```
EZD1851I  CSSMTP USER1 / TSU00050 on CSSMTP INPUT ERROR.
EDC8122I No buffer space available. (errno2=0x0000006B)
EZD1852I  jobname NO SPACE IN THE z/OS UNIX FILE SYSTEM MAIL DIRECTORY
```

**Explanation**
The Communications Server SMTP (CSSMTP) application cannot write to the configured or default z/OS UNIX file system mail directory associated with the ExtendedRetry statement because the directory is full or nearly full.

In the message text:

**jobname**
The job name of the task that started the CSSMTP application.

**System action**
The CSSMTP application stops.

**Operator response**
Contact the system programmer.

**System programmer response**
The extended retry function of the Communication Server SMTP (CSSMTP) application requires a UNIX file system mail directory to work correctly. Because the z/OS UNIX file system mail directory is full or nearly full, free some space in the directory or allocate a larger one.

Do one of the following tasks:

- Perform the following steps to free some space in the directory:
  1. Delete files in the z/OS UNIX file system that are not associated with the mail directory. If you need to delete mail files in the mail directory, you must delete both the .cf files and the .df files that are associated with the mail message. The mail associated with these files is lost.
  2. Restart the CSSMTP application.

- Perform the following steps to use a larger directory:
  1. Allocate a larger z/OS UNIX file system for the new mail directory.
  2. Copy the files from the old mail directory to the new mail directory.
  3. Change the MailDirectory parameter in the ExternalRetry configuration statement to specify the new directory.
  4. Restart the CSSMTP application.

See the ExtendedRetry statement in z/OS Communications Server: IP Configuration Reference for more information about the z/OS UNIX file system mail directory.
**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamlert

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you when CSSMTP has a problem storing mails into the z/OS UNIX file system mail directory.

**Example**

```
EZD1852I  CSSMTP1  NO SPACE IN THE z/OS UNIX FILE SYSTEM MAIL DIRECTORY
```

**EZD1856I  jobname  JES TASK SHORTAGE DETECTED**

**Explanation**
The Communications Server SMTP (CSSMTP) application detected that over 75% of DEST JES tasks or over 75% of WRITER JES tasks are waiting for long retry processing to complete for one or more mail messages in a JES spool file.

In the message text:

*jobname*

The job name of the task that started the CSSMTP application.

**System action**
Processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Determine whether you need to issue the MODIFY FLUSHRETRY or the MODIFY SUSPEND command.

1. Issue the MODIFY DISPLAY,SPoolStatus,Summary command to determine the number of mail messages that are in the pending or long retry state for each JES task. From the summary report, obtain the TKID value that has pending or long retry mail messages, then issue the MODIFY D,SPoolstatus,Detail,TKID= tkid
command to display detail information for the specific JES task. See the information about the CSSMTP MODIFY command information in z/OS Communications Server: IP System Administrator’s Commands for an example of the MODIFY DISPLAY,SPOOLSTATUS, command display.

2. Do one of the following:

- If there are few mail messages in long retry state for a large spool file, issue the MODIFY FLUSHRETRY command to force those long retry mail messages to be tried again immediately. If that retry fails, mail messages will become undeliverable mail. The JES task will then be freed to process other JES spool files.
- If there is a large number of mail messages in long retry state, then there might be a problem with the mail servers. Issue the MODIFY SUSPEND command to suspend new JES spool file processing until the problem is resolved. If message EZD1817I, EZD1818I, or EZD1819I is in the log, they might help to determine the problem. See EZD1817I, EZD1818I, or EZD1819I for more detail.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlhck

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when JES tasks become unavailable to process the JES spool file.

Example

| EZD1856I | CSSMTP1 JES TASK SHORTAGE DETECTED |

Explanation
The Communications Server SMTP (CSSMTP) application detected that less than 50% of DEST JES tasks and less than 50% of WRITER JES tasks are waiting for long retry processing to complete for one or more mail messages in a JES spool file. The previous JES tasks shortage is relieved.

In the message text:

jobname
The job name of the task that started the CSSMTP application.
System action
Processing continues.

Operator response
Contact the system programmer.

System programmer response
Determine whether you need to issue the MODIFY RESUME command.

1. Issue the MODIFY DISPLAY,SPOOLSTATUS,SUMMARY command to determine the state of the JES tasks. See the information about the CSSMTP MODIFY command in z/OS Communications Server: IP System Administrator’s Commands for an example of the MODIFY DISPLAY,SPOOLSTATUS command display.

2. If the state is SUPND, this indicates that a MODIFY SUSPEND command was issued to suspend the JES tasks. Issue a MODIFY RESUME command to resume the processing for new JES spool files.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlhck

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can alert you to when JES tasks are available to process new JES spool files.

Example

<table>
<thead>
<tr>
<th>EZD1857I  CSSMTP1  JES TASK SHORTAGE RELIEVED</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1858I  jobname  STORAGE SHORTAGE DETECTED IN THE addrJobName ADDRESS SPACE</td>
</tr>
</tbody>
</table>

Explanation
The Communications Server SMTP (CSSMTP) application detected that storage use in the CSSMTP application address space exceeds 75%.

In the message text:

jobname
The job name of the task that started the CSSMTP application.
addrJobName

The job name of the address space. This name matches the job name of the started CSSMTP application.

System action

Processing continues.

Operator response

Contact the system programmer.

System programmer response

Perform the following steps to determine the problem:

1. Issue the MODIFY D,SPOOLSTATUS,SUMMARY command to determine the number of mail messages that are in the pending or long retry state for each JES task. From the summary report, obtain the TKID value that has pending or long retry mail messages, then issue the MODIFY D,SPOOLSTATUS,DETAIL,TKID=\textit{tkid} command to display detail information for the specific JES task.

2. Do one of the following:
   - If there are few mail messages in long retry for a large spool file, you can issue the MODIFY FLUSHRETRY command to force those long retry mail messages to be tried again immediately. If that retry fails, mail messages will become undeliverable mail. The JES task will be freed to process other JES spool files and the large spool file will also be freed.
   - If there is a large number of mail messages in long retry, then there might be a problem with the mail servers. You can issue the MODIFY SUSPEND command to suspend new JES spool file processing until the problem is resolved. If messages EZD1817I, EZD1818I or EZD1819I are in the log, they might help you to determine the problem. See EZD1817I, EZD1818I, or EZD1819I for more detail.
   - If there are several large spool files in progress, then you can issue the MODIFY SUSPEND command to suspend new JES spool file processing until the storage problem is relieved.

User response

Not applicable.

Problem determination

See the system programmer response.

Source

z/OS Communications Server TCP/IP: CSSMTP

Module

ezamlhck

Routing code

10

Descriptor code

12
Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can enable you to automatically monitor storage use in the CSSMTP application address space.

Example

<table>
<thead>
<tr>
<th>EZD1858I CSSMTP1 STORAGE SHORTAGE DETECTED IN THE CSSMTP1 ADDRESS SPACE</th>
</tr>
</thead>
</table>

**EZD1859I**  
jobname STORAGE SHORTAGE RELIEVED IN THE addrJobname ADDRESS SPACE

Explanation
The Communications Server SMTP (CSSMTP) application detected that storage use in the CSSMTP application address space has dropped to less than 50%. The previous storage shortage is relieved.

In the message text:

**jobname**
The job name of the task that started the CSSMTP application.

**addrJobname**
The job name of the address space. This name matches the job name of the started CSSMTP application.

System action
Processing continues.

Operator response
Contact the system programmer.

System programmer response
Determine whether you need to issue the MODIFY RESUME command.

1. Issue the MODIFY D,SPOOLSTATUS,SUMMARY command to see the state of the JES tasks.
2. If the state is SUPND, this indicates that a MODIFY SUSPEND command was issued to suspend the JES tasks. Issue a MODIFY RESUME command to resume the processing for new JES spool files.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezankhck

Routing code
10
Automation

This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can enable you to automatically monitor storage use in the CSSMTP application address space.

Example

<table>
<thead>
<tr>
<th>EZD1859I</th>
<th>CSSMTP1 STORAGE SHORTAGE RELIEVED IN THE CSSMTP1 ADDRESS SPACE</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1860I</td>
<td>jobname STORAGE SHORTAGE DETECTED IN THE z/OS UNIX FILE SYSTEM DEAD LETTER DIRECTORY</td>
</tr>
</tbody>
</table>

Explanation

The Communications Server SMTP (CSSMTP) application detected that storage use in the z/OS UNIX file system dead letter directory exceeds 75%.

In the message text:

**jobname**

The job name of the task that started the CSSMTP application.

System action

Processing continues.

Operator response

Contact the system programmer.

System programmer response

A possible cause for this error is that the UNIX file system that contains the dead letter directory is full or nearly full. If this is the cause of the error, perform one of the following:

- Determine what can be deleted in the file system and delete it.
- Change the DeadLetterDirectory configuration statement to be a directory on a different file system and issue a MODIFY REFRESH command to use the new directory.

If the problem persists, then either issue a MODIFY SUSPEND command or change the DeadLetterAction configuration statement to DELETE.

Determine whether you need to issue the MODIFY SUSPEND command.

1. Issue the MODIFY D,SPOOLSTATUS,SUMMARY command to determine the number of mail messages that are in the pending or long retry state for each JES task. From the summary report, obtain the TKID value that has pending or long retry mail messages, then issue the MODIFY D,SPOOLSTATUS,DETAIL,TKID=tkid command to display detail information for the specific JES task.

2. If there is a large number of mail messages in long retry, then there might be a problem with the mail servers. Issue the MODIFY SUSPEND command to suspend new JES spool file processing until the problem is resolved. If messages EZD1817I, EZD1818I or EZD1819I are in the log, they might help in determining the problem. See EZD1817I, EZD1818I, or EZD1819I for more information.

3. Delete all unneeded dead letters to free some space in the z/OS UNIX file system dead letter directory.

See the information about the UNDELIVERABLE statement in z/OS Communications Server: IP Configuration Reference for more information about z/OS UNIX file system dead letter directory.
User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlhck

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can allow you to monitor storage use in the z/OS UNIX file system dead letter directory.

Example

<table>
<thead>
<tr>
<th>Message Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1860I</td>
<td>CSSMTP1 STORAGE SHORTAGE DETECTED IN THE z/OS UNIX FILE SYSTEM DEAD LETTER DIRECTORY</td>
</tr>
<tr>
<td>EZD1861I</td>
<td>jobname STORAGE SHORTAGE RELIEVED IN THE z/OS UNIX FILE SYSTEM DEAD LETTER DIRECTORY</td>
</tr>
</tbody>
</table>

Explanation
The Communications Server SMTP (CSSMTP) application detected that storage use in the z/OS UNIX file system dead letter directory has dropped to less than 50%. The previous storage shortage is relieved.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
Processing continues.

Operator response
Contact the system programmer.

System programmer response
Issue the MODIFY D,SPOOLSTATUS,SUMMARY command to display the state of the JES task. If the JES task state is SUPND, it indicates that a MODIFY SUSPEND command was issued to suspend the JES task. Issue a MODIFY RESUME command to resume processing the new JES spool files.
User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CSSMTP

Module
ezamlhck

Routing code
10

Descriptor code
12

Automation
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can allow you to monitor storage use in the z/OS UNIX file system dead letter directory.

Example
EZD1861I CSSMTP1 STORAGE SHORTAGE RELIEVED IN THE Z/OS UNIX FILE SYSTEM DEAD LETTER DIRECTORY

EZD1862I  jobname STORAGE SHORTAGE DETECTED IN THE z/OS UNIX FILE SYSTEM MAIL DIRECTORY

Explanation
The Communications Server SMTP (CSSMTP) application detected that storage use in the z/OS UNIX file system mail directory associated with the ExtendedRetry statement exceeds 75%.

In the message text:

jobname
The job name of the task that started the CSSMTP application.

System action
Processing continues.

Operator response
Contact the system programmer.

System programmer response
This error indicates that the UNIX file system that contains the mail directory is nearly full. If the mail directory reaches this condition, the CSSMTP application will stop. To prevent CSSMTP from stopping, perform one of the following actions:

- Delete any files in the file system that you no longer need. Do not delete any files from the mail directory.
- If target servers are available and receiving mail, use the MODIFY FLUSHRETRY,AGE=days operator command so that older mail messages can be processed from the mail directory.
If the problem persists, you can perform the following steps to stop the CSSMTP application and move the mail directory to a different file system with more space.

1. Use the `mkdir` command to define the new directory.
2. Copy the existing files from the old directory to the new directory.
3. Update the MailDirectory parameter on the ExtendedRetry configuration statement to specify the new directory.
4. Restart the CSSMTP application.

If the storage shortage for the z/OS Unix file system that contains the mail directory is relieved, message EZD1863I is displayed.

See ExtendedRetry statement in z/OS Communications Server: IP Configuration Reference for more information about the z/OS UNIX file system mail directory.

**User response**

Not applicable.

**Problem determination**

If you need to determine whether the target servers are having problems receiving the mail, see gathering diagnostic information about CSSMTP problems in z/OS Communications Server: IP Diagnosis Guide for setting up diagnostic traces.

**Source**

z/OS Communications Server TCP/IP: CSSMTP

**Module**

ezamhlck

**Routing code**

10

**Descriptor code**

12

**Automation**

This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can allow you to monitor storage use in the z/OS UNIX file system mail directory.

**Example**

```
EZD1862I  CSSMTP1 STORAGE SHORTAGE DETECTED IN THE z/OS UNIX FILE SYSTEM MAIL DIRECTORY
```

```
EZD1863I  jobname STORAGE SHORTAGE RELIEVED IN THE z/OS UNIX FILE SYSTEM MAIL DIRECTORY
```

**Explanation**

The Communications Server SMTP (CSSMTP) application detected that storage use in the z/OS UNIX file system mail directory has dropped to less than 50%. The storage shortage indicated by the preceding message EZD1862I is relieved.

In the message text:
**jobname**
The job name of the task that started the CSSMTP application.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: CSSMTP

**Module**
ezamlhck

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is written to the system console and log file for CSSMTP. This message is a good candidate for automation. Automation can allow you to monitor storage use in the z/OS UNIX file system mail directory.

**Example**

```
EZD1863I CSSMTP1 STORAGE SHORTAGE RELIEVED IN THE Z/OS UNIX FILE SYSTEM MAIL DIRECTORY
```

EZD1901I Remote security endpoint at `remote_ip` port `remote_port` is using an unsupported authentication method `auth_method_string` `auth_method`

**Explanation**
The authentication method that is identified in the authentication payload that was sent by the remote security endpoint is not supported. The Internet Key Exchange (IKE) daemon cannot authenticate the remote security endpoint.

In the message text:

**remote_ip**
The remote security endpoint IP specification.
remote_port
The remote port of the IKE daemon peer.

auth_method_string
The string that identifies the authentication method being used by the peer.

auth_method
The decimal value that represents the authentication method that is used by the peer.

System action
The IKE SA negotiation fails; IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Notify the administrator of the remote security endpoint that it is using an unsupported authentication method.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2IKEAuthRequest.cpp, IKEv2IKEAuthResponse.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1901I Remote security endpoint at 1.2.3.4 port 500 is using an unsupported authentication method DSS_Digital_Signature 3

EZD1902I Delayed rekey attempt for IKE tunnel tunnel_id; proceeding with incomplete exchanges: half_open_count half-open, half_closed_count half-closed, rekey_requested_count rekeying

Explanation
The Internet Key Exchange (IKE) daemon delayed a rekey attempt for the specified IKE tunnel to allow exchanges for one or more of its child Security Associations (SAs) to complete. The limit for the length of the delay was reached, so the rekey attempt has been initiated.
In the message text:

`tunnel_id`

The tunnel prefix and number that were used to identify the tunnel.

`half_open_count`

The count of child SAs that are in HALF_OPEN state at the time that the IKE SA rekey attempt was initiated.

`half_closed_count`

The count of child SAs that are in HALF_CLOSED state at the time that the IKE SA rekey attempt was initiated.

`rekey_requested_count`

The count of child SAs that are in the process of being rekeyed at the time that the IKE SA rekey attempt was initiated.

**System action**

IKE daemon processing continues

**Operator response**

None.

**System programmer response**

None.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: IKE daemon

**Module**

IKEv2IKESA.cpp

**Routing code**

11

**Descriptor code**

7

**Automation**

This message is output to syslog

**Example**

```
EZD1902I Delayed rekey attempt for IKE tunnel K21; proceeding with incomplete exchanges: 0 half-open, 1 half-closed, 0 rekeying

EZD1903I  A request_type request message sent to the NSS server with correlator ID corr_id timed out
```

1254 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Explanation
The Internet Key Exchange (IKE) daemon did not receive a timely response from the network security services (NSS) server for the corresponding request.

In the message text:

**request_type**
The type of request to the NSS server that timed out.

**corr_id**
The 16-byte message correlator contained in the request message.

System action
The IKE SA negotiation fails; IKE daemon processing continues.

Operator response
None.

System programmer response
Verify that the NSS server is operational, and that the network that connects it to the IKE daemon is not congested.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2AuthRequest.cpp, IKEv2AuthResponse.cpp, and phase1.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example

EZD1903I A NSS_CreateSignatureReqToSrv request message sent to the NSS server with correlator ID 00000000000000000000000000000000 timed out

EZD1904E IKED NSS client API version clientAPIversion does not match NSS server API version serverAPIversion for stack stackname
Explanation
The Internet Key Exchange daemon (IKED) connected to a Network Security Services (NSS) server that was supporting a lower level of the NSS API. Certificate services such as Certificate Revocation List (CRL) checking, Certificate Authority (CA) Hierarchy validation, and HTTP certificate retrieval are not available for this stack. The lack of this support limits the kinds of tunnels that the IKE daemon can negotiate for the stack. The IKE daemon is able to negotiate IKEv1 and IKEv2 tunnels using preshared key authentication for the stack. The IKE daemon is able to negotiate IKEv1 tunnels with digital signature authentication but without support for CRL checking or CA Hierarchy validation for the stack. The IKE daemon is unable to negotiate IKEv2 tunnels with digital signature authentication for the stack.

In the message text:

- **clientAPIversion**
  - The version of the NSS client API supported by the IKE daemon.

- **serverAPIversion**
  - The version of the NSS server API supported by the NSS server.

- **stackname**
  - The name of the stack that connected to the NSS server supporting a lower level of the NSS API.

System action
IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Connect the IKE daemon NSS client to an NSS server that is running on a z/OS Communications Server V1R12 or greater system. To identify the NSS server that the IKE daemon is currently connected to, inspect the system log for EZD1136I.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
stackobj.cpp

Routing code
3

Descriptor code
3

Automation
This message goes to the console and to syslog.
EZD1904E IKED NSS client API version 4 does not match NSS server API version 1 for stack TCPCS

EZD1905I IKE version version tunnel activation for stackname using KeyExchangeRule kername requires the NSS certificate service but the service is not available - return code = retcode

Explanation
The Internet Key Exchange (IKE) daemon is unable to activate an IKE tunnel because the required NSS certificate service is not available.

In the message text:
version
- The version of the IKE protocol being used to activate the tunnel.

stackname
- The name of the TCP/IP stack for which the IKE tunnel is being activated.

kername
- The name of the KeyExchangeRule being used for this IKE tunnel activation attempt.

retcode
- The reason why the required certificate service is unavailable. Possible values are:
  -1 The stack is not configured as an NSS client.
  -2 The stack is configured as an NSS client but is not configured for the certificate service.
  -3 The stack is connected to an NSS server but the stack is not authorized to use the certificate service.
  -4 The stack is connected to an NSS server that does not support advanced PKI certificate services.

The NSS certificate service is required for all IKEv2 tunnel activation requests that use an authentication method other than PresharedKey. For example, if the KeyExchangeAction statement for the specified KeyExchangeRule has HowToInitiate IKEv2 and HowToAuthMe RSASignature, the NSS certificate service is required to activate the IKE tunnel.

System action
This tunnel activation attempt fails. IKE daemon processing continues.

Operator response
None.

System programmer response
The appropriate corrective action depends on the retcode value in the message:
-1 Configure the stack as an NSS client that requests NSS certificate services.
-2 Configure the stack as an NSS client that requests NSS certificate services.
-3 Notify the system programmer of the NSS server to provide authorization to the stack for network security certificate services.
-4 Change the configuration of the IKE daemon to connect to an NSS server that does support advanced PKI certificate services; for example, an NSS server on a z/OS V1R12 system.
See the information about IP security in z/OS Communications Server: IP Configuration Guide for information about network security certificate services.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
anchor_ureq.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to syslog.

**Example**

```
EZD1905I  IKE version 2.0 tunnel activation for TCPCS2 using KeyExchangeRule IKEV2-SA1-TCP requires the NSS certificate service but the service is not available - return code = -1
```

**EZD1906I**  FIPS140 support is not enabled for the IKE daemon

**Explanation**
The Federal Information Processing Standard 140 (FIPS 140) function is not enabled for the IKE daemon. Cryptographic operations might be performed by cryptographic modules that do not follow the Level 1 security requirements of Federal Information Processing Standard (FIPS) publication 140-2.

**System action**
IKE daemon processing continues.

**Operator response**
None

**System programmer response**
If FIPS 140 support is required for the IKE daemon, then configure **FIPS140 Yes** on the IkeConfig statement in the IKED configuration file; otherwise, no action is needed.

**User response**
Not applicable.
Problem determination
None

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1906I FIPS140 support is not enabled for the IKE daemon

EZD1907I FIPS140 support is enabled for the IKE daemon

Explanation
The Federal Information Processing Standard 140 (FIPS 140) function is enabled for the IKE daemon. All cryptographic operations are performed by cryptographic modules that are designed to follow the Level 1 security requirements of Federal Information Processing Standard (FIPS) publication 140-2.

System action
IKE daemon processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon
Module
ike_config.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1907I FIPS140 support is enabled for the IKE daemon

EZD1908I Tunnel activation for stackname using KeyExchangeRule kername failed because the identity type is not compatible with the authentication method

Explanation
An attempt to initiate Internet Key Exchange version 2 (IKEv2) tunnel activation for a stack failed because the LocalSecurityEndpoint statement that is defined for the KeyExchangeRule statement has Identity type KeyId, but the KeyExchangeAction statement specifies a local authentication method on the HowToAuthMe parameter that is not pre-shared key. Identity type KeyId can be used only in conjunction with PresharedKey authentication.

In the message text:

stackname
The name of the stack for which the IKE tunnel is being activated

kername
The name of the KeyExchangeRule for this IKE tunnel activation attempt.

System action
This tunnel activation attempt fails. IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Find the named KeyExchangeRule in the IPSec policy definitions and change the HowToAuthMe value to PresharedKey, or change the LocalSecurityEndpoint Identity to a type that is compatible with the authentication method specified on HowToAuthMe value (for example, Fqdn).

User response
Not applicable.

Problem determination
Not applicable.
Source
z/OS Communications Server TCP/IP: IKE daemon

Module
anchor_ureq.cpp

Routing code
7

Descriptor code
11

Automation
The message is output to syslog

Example
EZD1908I Tunnel activation for TCPCS2 using KeyExchangeRule IKEv2-SA1-TCP failed because the identity
type

is not compatible with the authentication method

EZD1909I   IP validation failed: the remote identity peer_id does not match remote IP address
 peer_ip_addr

Explanation
Local policy required that the IP type identity of the internet key exchange (IKE) peer be validated by comparing
it to the IP address of the IKE peer. The IP validation failed because the remote identity received from the IKE
peer does not match the IP address of the IKE peer.

Additional diagnostic messages with the same message instance number will be issued to identify the impacted
Security Association (SA). The message instance number precedes the message number in the log output and is
used to group related messages from the IKE daemon.

In the message text:
peer_id
   The identity of the IKE peer.
peer_ip_addr
   The IP address of the IKE peer.

System action
The SA negotiation fails. IKE daemon processing continues.

Operator response
Contact the system programmer

System programmer response
Locate the KeyExchangeRule statement in the IP Security (IPSec) policy definitions associated with the
impacted SA. Set the ByPassIPValidation parameter to yes in the associated KeyExchangeAction statement to
avoid the IP validation check or change the associated RemoteSecurityEndpoint Identity parameter to include
the remote peer IP address. The IP validation check can be overridden globally by using the ByPassIPValidation
parameter on the KeyExchangePolicy statement in the IPSec policy. The ByPassIPValidation parameter should
be set to yes if the RemoteSecurityEndpoint peer is behind a network address translation (NAT) device.
See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
CommonDomainOfInterpretation.cpp

**Routing code**
*

**Descriptor code**
*

**Automation**
The message is output to syslog

**Example**

```
EZD1909I  IP validation failed: the remote identity 10.83.2.4 does not match remote IP address 10.84.2.4
EZD1909I  IP validation failed: the remote identity 2001:db8:10::83:2:2 does not match remote IP address 2001:db8:10::84:2:2
```

**EZD1910I**  FIPS140 support is enabled for the IKE daemon and no valid KeyExchangeOffers were found in KeyExchangeAction (KEAname)

**Explanation**
This message is issued when the IKE daemon is enabled to support the Level 1 security requirements of Federal Information Processing Standard publication 140-2 (FIPS 140), and one or more KeyExchangeOffer objects were omitted from the specified KeyExchangeAction object. If the IKE daemon is enabled for FIPS 140, the daemon omits KeyExchangeOffer objects that use the DES, MD5, or AES_XCBC cryptographic algorithms, or Diffie-Hellman groups 1, 2, or 5 from any proposal it builds.

In the message text:

**KEAname**
The KeyExchangeAction name that is configured in the policy.

**System action**
The SA negotiation fails; the IKE daemon continues.

**Operator response**
Contact the system programmer.
System programmer response

If you want the IKE daemon to be enabled to support FIPS 140, ensure that at least one KeyExchangeOffer object exists in the specified KeyExchangeAction object that does not contain any of the following:

- HowToEncrypt DES
- HowToAuthMsgs MD5
- HowToVerifyMsgs HMAC_MD5_96
- HowToVerifyMsgs AES128_XCBC_96
- PseudoRandomFunction HMAC_MD5
- PseudoRandomFunction AES128_XCBC
- DHGroup Group1, Group2, Group5

If you do not want to continue to have the IKE daemon enabled to support FIPS 140, then configure FIPS140 No on the IkeConfig statement in the IKED configuration file and restart the IKE daemon.

See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP: IKE daemon

Module

config_adapter.cpp

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

Example

EZD1910I FIPS140 support is enabled for the IKE daemon and no valid KeyExchangeOffers were found in KeyExchangeAction ( TCS4_Vipa81-TCS7_Vipa81 )

EZD1911I FIPS140 support is enabled for the IKE daemon and no valid IpDataOffers were found in IpDynVpnAction ( IDVAname )

Explanation

This message is issued when the IKE daemon is enabled to support the Level 1 security requirements of Federal Information Processing Standard publication 140-2 (FIPS 140), and one or more IpDataOffer objects were omitted from the specified IpDynVpnAction object. If the IKE daemon is enabled for FIPS 140, the daemon
omits IpDataOffer objects that use the DES, MD5, or AES_XCBC cryptographic algorithms, or Pfs with Diffe-Hellman groups 1, 2, or 5 from any proposal it builds.

In the message text:

**IDVAname**
The IpDynVpnAction name that is configured in the policy.

**System action**
The SA negotiation fails; the IKE daemon continues.

**Operator response**
Contact the system programmer.

**System programmer response**
If you want the IKE daemon to be enabled to support FIPS 140, ensure that at least one IpDataOffer object exists in the specified IpDynVpnAction object that does not contain any of the following:

- HowToEncrypt DES
- HowToAuth Hmac_MD5
- HowToAuth AES128_XCBC_96
- Pfs Group1, Group2, or Group5 (specified in the IpDataOffer's associated IpDynVpnAction)
- InitiateWithPfs Group1, Group2, Group5 (specified in the IpDataOffer's associated IpDynVpnAction)
- AcceptablePfs Group1, Group2, Group5 (specified in the IpDataOffer's associated IpDynVpnAction)

If you do not want the IKE daemon to be enabled to support FIPS 140, then configure FIPS140 No on the IkeConfig statement in the IKED configuration file and restart the IKE daemon.

See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
policy.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.
Example

EZD1911I FIPS140 support is enabled for the IKE daemon and no valid IpDataOffers were found in IpDynVpnAction ( TCS4_Vipa81-TCS7_Vipa81 )

EZD1912I  No SharedKey was specified in KeyExchangeRule (KERname) and KeyExchangeOffers in KeyExchangeAction (KEAname) require PresharedKey authentication

Explanation

One or more of the KeyExchangeOffer objects that are available for use by the specified KeyExchangeRule object was configured to use PresharedKey to authenticate peers, but the KeyExchangeRule object did not specify a SharedKey to be used.

In the message text:

KERname  The KeyExchangeRule name that is configured in the policy.

KEAname  The KeyExchangeAction name that is configured in the policy.

System action

The SA negotiation fails; the IKE daemon continues.

Operator response

Contact the system programmer.

System programmer response

If you want to use a pre-shared key to authenticate peers during this SA negotiation, specify a pre-shared key on the SharedKey parameter in the associated KeyExchangeRule object. Otherwise, you can use a digital signature to authenticate peers by specifying a digital signature authentication method on the HowToAuthPeers parameter in one or more of the KeyExchangeOffer objects associated with this KeyExchangeRule object. See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP: IKE daemon

Module

config_adapter.cpp

Routing code

Not applicable.

Descriptor code

Not applicable.
Automation
Not applicable.

Example

EZD1912I No SharedKey was specified in KeyExchangeRule ( KER:TCS4_Vipa81-TCS7_Vipa81 ) and KeyExchangeOffers in KeyExchangeAction ( KEA:TCS4_Vipa81-TCS7_Vipa81 ) require PresharedKey authentication

EZD1913I ICSF callable service routine failed RC: return_code RSN: reason_code

Explanation
An Integrated Cryptographic Service Facility (ICSF) callable service that is required by the IKE daemon has failed.

In the message text:

routine
The ICSF callable service.

return_code
The failure code returned by ICSF.

reason_code
The failure reason code returned by ICSF.

System action
The IKE function that called the ICSF callable service fails; the IKE daemon continues.

Operator response
Contact the system programmer.

System programmer response
Examine the syslog for subsequent failures to determine the consequences. See the information about the ICSF and TSS Return and Reason Codes in z/OS Cryptographic Services ICSF Application Programmer's Guide to determine the reason for the failure and possible corrective actions.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
dh_ecp_module.cpp

Routing code
Not applicable for syslog message.
**Example**

EZD1913I  ICSF callable service CSFPGKP failed RC: 0000000C RSN: 00000000

Return Code (C) with Reason Code (0) indicates that ICSF is not available.

EZD1914I  Remote security endpoint at `remote_ip` port `remote_port` sent a signing certificate with encoding `encoding` that is not allowed

**Explanation**

An IKE version 2.0 Security Association (SA) activation attempt failed because the remote security endpoint sent a signing certificate that contained encoding that is not allowed by locally defined IPSec policy. The signing certificate appears in the first certificate payload.

In the message text:

- `remote_ip`: The remote security endpoint IP specification.
- `remote_port`: The port of the remote security endpoint.
- `encoding`: The encoding of the received signing certificate payload.

**System action**

The IKE SA negotiation fails; IKE daemon processing continues.

**Operator response**

None.

**System programmer response**

Notify the administrator of the remote security endpoint that it must not send certificate payloads that contain the disallowed encoding. Alternatively, the administrator of the z/OS security endpoint can change local IPSec policy to allow such certificate payload encodings.

See the information about Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for information about the CertificateURLLookupPreference keyword of the KeyExchangePolicy or KeyExchangeAction statement.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP: IKE daemon
Module
IKEv2AuthRequest.cpp, IKEv2AuthResponse.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example
EZD1914I Remote security endpoint at 1.2.3.4 port 500 sent a signing certificate with encoding 12 that is not allowed

EZD1915I Rekey attempt for IKE tunnel tunnel_id rejected because of incomplete exchanges:
 half_open_count half-open, half_closed_count half-closed, rekey_requested_count rekeying

Explanation
The IKE daemon received a rekey request for the specified IKE tunnel from the IKE peer. The request is rejected to allow incomplete exchanges for one or more of the IKE tunnel's child Security Associations (SAs) to complete before replacing the IKE SA with a new IKE SA.

IKE rejects the rekey request by sending a NO_PROPOSAL_CHOSEN notification to the peer.

In the message text:
- **tunnel_id**
  - The tunnel prefix and number used to identify the tunnel.
- **half_open_count**
  - The count of child SAs that are in HALF_OPEN state.
- **half_closed_count**
  - The count of child SAs that are in HALF_CLOSED state.
- **rekey_requested_count**
  - The count of child SAs that are in the process of being rekeyed.

System action
The IKE tunnel rekey request fails. IKE daemon processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.
Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
IKEv2CreateChildRequest.cpp

Routing code
*

Descriptor code
*

Automation
Not applicable.

Example
EZD1915I Rekey attempt for IKE tunnel K21 rejected because of incomplete exchanges: 1 half-open, 0 half-closed, 0 rekeying

EZD1916I NSS server cryptographic services are disabled for stack tcpname - FIPS140 support is enabled for the IKE daemon but is not enabled for the NSS server

Explanation
The Federal Information Processing Standard 140 (FIPS 140) function is enabled for the IKE daemon, but it is not enabled for the network security services (NSS) server. The NSS server is not permitted to provide cryptographic services to the IKE daemon for the stack.

In the message text:

tcpname
    The name of the affected TCP/IP stack.

System action
IKED will not enable cryptographic services through the NSS server for the named stack. IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
When the IKE daemon is enabled for FIPS 140 support, stacks that are configured to use the NSS server for cryptographic services require that the NSS server also be enabled for FIPS 140 support.

The stack configuration, the IP security policy for the stack, the IKE daemon configuration, and the NSS server configuration must all be consistent. To understand the implications and requirements for enabling FIPS 140 support in your environment, see the information about FIPS 140 and IP security in z/OS Communications Server: IP Configuration Guide.
User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
stackobj.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example

EZD1916I NSS server cryptographic services are disabled for stack TCPCS - FIPS140 support is enabled for the IKE daemon but is not enabled for the NSS server.

EZD1917I IKE status for stack tcpname is FIPS140 enabled but IKED is not FIPS140 enabled

Explanation
The Federal Information Processing Standard 140 (FIPS 140) function is enabled for the named TCP/IP stack, but it is not enabled for the IKE daemon. The Internet Key Exchange (IKE) daemon is not permitted to provide cryptographic services to the stack.

In the message text:

tcpname
The name of the affected TCP/IP stack.

System action
The IKE daemon will not perform Security Association (SA) negotiation or any other cryptographic services for the specified stack. IKE daemon processing continues.

Operator response
Contact the system programmer.

System programmer response
Stacks that are enabled for FIPS 140 support require that the IKE daemon also be enabled for FIPS 140 support. The stack configuration, the IP security policy for the stack, the IKE daemon configuration, and the NSS server configuration must all be consistent. To understand the implications and requirements for enabling FIPS 140 support in your environment, see the information about FIPS 140 and IP security in z/OS Communications Server: IP Configuration Guide.
User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
stackobj.cpp

Routing code
11

Descriptor code
7

Automation
This message is output to syslog.

Example

<table>
<thead>
<tr>
<th>EZD1917I</th>
<th>IKE status for stack TCPCS is FIPS140 enabled but IKED is not FIPS140 enabled.</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1918I</td>
<td>A cryptographic key in use is too short for the chosen Auth or PRF algorithm when FIPS140 is enabled: key length <code>keylen</code> bytes, minimum required <code>minlen</code> bytes</td>
</tr>
</tbody>
</table>

Explanation
When Federal Information Processing Standard publication 140 (FIPS 140) support is enabled for the IKE daemon, all of the cryptographic keys that are used by the chosen authentication (Auth) or pseudo random function (PRF) algorithm must be at least half the length of the PRF digest size. These cryptographic keys can be the configured pre-shared key that is used for IKE authentication, or, if you are using Internet Key Exchange version 2 (IKEv2), they can be the keys that are used by the IKE daemon to internally generate keying material for a prior IKE SA.

In the message text:

`keylen`
The length of the key.

`minlen`
The minimum key length that is required for the chosen Auth or PRF algorithm.

System action
IKED phase 1 tunnel negotiation fails. IKE daemon processing continues.

Operator response
Contact the system programmer.
System programmer response
Examine the surrounding IKED messages in the syslogd log file to determine which tunnel is affected. The following criteria apply when FIPS 140 support is enabled for the IKE daemon:

- If a pre-shared key is configured, it must be at least half as long as the key used by the configured Auth and PRF algorithms. If the pre-shared key is not that length, the IKED phase 1 tunnel negotiation fails, and message EZD1918I is issued. To prevent the negotiation from failing, modify the tunnel policy to configure a longer pre-shared key, or modify the tunnel policy to use an Auth or PRF algorithm with a shorter key. Policy changes must be coordinated across all endpoints that are involved in the tunnel negotiation.

- You should not modify the policy for an active IKEv2 tunnel to specify a PRF algorithm that uses a key that is more than twice the length of the originally specified PRF algorithm. You need to deactivate the IKE tunnel before you make such a modification; otherwise, a refresh of the active tunnel might fail and message EZD1918I will be issued. For example, if you switch from the HMAC_SHA1 algorithm to the HMAC_SHA2_256 algorithm, message EZD1918I will be issued during the tunnel refresh, but if you switch from the HMAC_SHA2_256 algorithm to the HMAC_SHA1 algorithm, the message will not be issued.

See FIPS 140 and IP security in z/OS Communications Server: IP Configuration Guide for information about FIPS 140 support in your environment.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
icsf_hmac.cpp, IKEv2IKESAKEP.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>EZD1918I</th>
<th>A cryptographic key in use is too short for the chosen Auth or PRF algorithm when FIPS140 is enabled: key length 12 bytes, minimum required 16 bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1919I</td>
<td>FIPS140 support is enabled for the IKE daemon but it has read access to CRYPTOZ resource FIPSEXEMPT.SYSTOK-SESSION-ONLY</td>
</tr>
</tbody>
</table>

Explanation
When the IKE daemon is configured for Federal Information Processing Standard 140 (FIPS 140) mode, the IKE daemon must have no access privileges (NONE) to the SAF resource FIPSEXEMPT.SYSTOK-SESSION-ONLY in the CRYPTOZ class.
See the FIPS140 parameter in z/OS Communications Server: IP Configuration Reference for information about configuring FIPS 140.

**System action**
The IKE daemon ends.

**Operator response**
Contact the system programmer.

**System programmer response**
Remove IKE daemon access to the SAF resource, or disable FIPS 140 mode for the IKE daemon. See the FIPS140 parameter in z/OS Communications Server: IP Configuration Reference for about configuring FIPS 140.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
ike_config.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is written to the MVS console and to syslog. This message is a good candidate for automation.

**Example**
Not applicable.

**EZD1920I**
Attempting an on-demand activation for IKE outbound UDP traffic from source_ipaddr port source_port to dest_ipaddr port dest_port using anchor filter filtername

**Explanation**
The Internet Key Exchange (IKE) daemon is attempting to negotiate an on-demand IPsec Security Association to protect outbound traffic on UDP ports 500 or 4500. The IKE daemon might use either port 500 or port 4500 when it is negotiating Security Associations.

This is not an error condition, but it is a possible indication of a misconfiguration. Usually, IKE daemon UDP traffic is allowed without IPsec protection. If the IKE daemon must negotiate a Security Association to protect its own messages, then it is likely that the negotiation will fail. If the negotiation fails, the IKE daemon issues subsequent syslog messages to indicate that the Security Association negotiation failed.
See the steps for configuring IP security policy in z/OS Communications Server: IP Configuration Guide for information.

In the message text:

**source_ipaddr**
The source IP address.

**source_port**
The source port.

**dest_ipaddr**
The destination IP address.

**dest_port**
The destination port.

**filtername**
The name of matching anchor filter rule.

**System action**
The IKE daemon continues processing.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
anchor_ureq.cpp

**Routing code**
*

**Descriptor code**
*

**Automation**
Not applicable.

**Example**

EZD1920I Attempting an on-demand activation for IKE outbound UDP traffic from 8.1.1.1 port 500 to 9.2.2.2 port 500 using anchor filter tunnel_1
EZD1921I Certificate ( label ) contains a key that is too short for FIPS 140 mode

Explanation
The Internet Key Exchange (IKE) daemon is configured to run in a mode that supports Federal Information Processing Standard 140 (FIPS 140). The IKE daemon detected that a certificate with the specified label contains an RSA key that is not allowed by FIPS 140. The certificate will not be available for IKE RSA mode authentication.

See the information about FIPS 140 and IP security in z/OS Communications Server: IP Configuration Guide.

In the message text:

label
   The label of the certificate.

System action
IKE processing continues.

Operator response
None.

System programmer response
If FIPS 140 support is required and the certificate is required for the RSA mode authentication, re-key the certificate with an RSA key that has a key size of 1024 bits or greater. If FIPS 140 support is not required for the IKE daemon, stop the daemon, configure FIPS140 No in the IKE configuration file, and restart the daemon.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
certcache.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1921I Certificate ( Certificate512 ) contains a key that is too short for FIPS 140 mode
EZD1922I  IKE STATUS FOR STACK stackname IS ACTIVE WITHOUT PORTS

Explanation
The IKE daemon detected that the specified stack is active, but it was not able to complete initialization of ports 500 and 4500. The IKE daemon cannot establish security associations for the specified stack, because port initialization did not succeed.

In the message text:

stackname
   The name of the stack that is active without ports.

System action
The IKE daemon continues.

Operator response
Contact the system programmer.

System programmer response
Examine the syslogd output file for errors preceding message EZD1922I. The most likely cause is failing to bind UDP port 500 or UDP port 4500, because the port is in use by another application or reserved for another application. If so, check the PORT statement in the TCPIP profile for the specified stack. Ensure that UDP port 500 and UDP port 4500 are reserved for the IKE daemon. If some other internal errors occur, contact the IBM support center.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
stackobj.cpp

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example

EZD1922I IKE STATUS FOR STACK TCPCS IS ACTIVE WITHOUT PORTS
EZD1923I  A create signature request to the NSS server failed; no matching certificate was found for local security endpoint identity local_id and authentication method auth_method

Explanation
The IKE daemon made a network security certificate services request to create a digital signature. A certificate matching the local security endpoint identity and authentication method was not found on the network security services (NSS) server.

In the message text:

local_id
The local security endpoint identity.

auth_method
The authentication method.

System action
The request fails; the IKE daemon continues.

Operator response
Contact the system programmer.

System programmer response
If you want to use network security certificate services with the specified local security endpoint identity, add a matching certificate to the key ring of the NSS server. See the Steps for configuring the NSS server in z/OS Communications Server: IP Configuration Guide for more information.

User response
Not applicable.

Problem determination
Not Applicable.

Source
z/OS Communications Server TCP/IP: IPSec

Module
CommonIKESA.cpp

Routing code
10

Descriptor code
12

Automation
This message is output to syslog.
Example

EZD1923I A create signature request to the NSS server failed; no matching certificate was found for local security endpoint identity 1.2.3.4 and authentication method RsaSignature

EZD1924I IKE detected a NAT while initiating a new tunnel mode IKEv2 dynamic tunnel with a non-z/OS peer

Explanation

The Internet Key Exchange (IKE) daemon is initiating a tunnel-mode Security Association (SA) for a new IKEv2 dynamic tunnel with a non-z/OS peer. The SA traverses a Network Address Translation (NAT) device. There might be problems with interoperability with the non-z/OS peer for a tunnel-mode SA. z/OS is providing NAT traversal support for a defined group of configurations where z/OS is running the IKE daemon. See the IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations and interoperability considerations.

System action

The SA negotiation continues.

Operator response

If the SA negotiation fails or if data cannot be successfully sent over the SA, contact the system programmer.

System programmer response

Determine whether there is an interoperability concern that caused the SA negotiation or data flow to fail. See the IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations and interoperability considerations.

If this is a host-to-host tunnel, a possible solution is to use a transport-mode IpDynVpnAction object instead of a tunnel-mode IpDynVpnAction object. See the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP: IKE daemon

Module

CommonIPsecSA.cpp

Routing code

2

Descriptor code

5
Automation
Not Applicable.

Example

EZD1924I IKE detected a NAT while initiating a new tunnel mode IKEv2 dynamic tunnel with a non-z/OS peer

EZD1925I IKE detected a NAT while initiating a new transport mode IKEv2 dynamic tunnel with a non-z/OS peer

Explanation
The Internet Key Exchange (IKE) daemon is initiating a transport-mode Security Association (SA) for a new IKEv2 dynamic tunnel with a non-z/OS peer. The SA traverses a Network Address Translation (NAT) device. There might be problems with interoperability with the non-z/OS peer for a transport-mode SA. z/OS is providing NAT traversal support for a defined group of configurations where z/OS is running the IKE daemon. See the IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations and interoperability considerations.

System action
The SA negotiation continues.

Operator response
If the SA negotiation fails or if data cannot be successfully sent over the SA, contact the system programmer.

System programmer response
Determine whether there is an interoperability concern that caused the SA negotiation or data flow to fail. See the IP security in z/OS Communications Server: IP Configuration Guide for a description of the supported configurations and interoperability considerations. Confirm that the non-z/OS peer supports transport-mode with NAT traversal as defined in RFC 5996 section 2.23.1.

A possible solution is to use a tunnel-mode IpDynVpnAction object instead of a transport-mode IpDynVpnAction object. See the Policy Agent and policy applications in z/OS Communications Server: IP Configuration Reference for more information about configuring policy.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: IKE daemon

Module
CommonIPsecSA.cpp

Routing code
2
Descriptor code
5

Automation
Not applicable.

Example

EZD1925I IKE detected a NAT while initiating a new transport mode IKEv2 dynamic tunnel with a non-
z/OS peer

EZD1926I CONTROL CONNECTION TO DESTINATION IP ADDRESS ip_addr ESTABLISHED

Explanation
The control connection to the non-z/OS target was established. The non-z/OS target is available for distributed
connections.

In the message text:

ip_addr
  The IP address of a distributed non-z/OS target.

System action
TCP/IP continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Sysplex Distributor

Module
EZBXFSUB

Routing code
2,8

Descriptor code
12
Automation
This message is written to the system console. This message is a good candidate for automation. Automation can allow you to monitor the status of sysplex distributor control connections to non-z/OS targets.

Example

```
EZD1926I CONTROL CONNECTION TO DESTINATION IP ADDRESS 9.42.105.53 ESTABLISHED
```

```
EZD1927I  CONTROL CONNECTION TO DESTINATION IP Address ip_addr IS CLOSED
```

Explanation
The control connection to the non-z/OS target closed because it is no longer a target for any distributed connections.

In the message text:

`ip_addr`
The IP address of a distributed non-z/OS target.

System action
TCP/IP continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Sysplex Distributor

Module
EZBXFSUB

Routing code
2,8

Descriptor code
12

Automation
This message is written to the system console. This message is a good candidate for automation. Automation can allow you to monitor the status of sysplex distributor control connections to non-z/OS targets.
Explanation
The TCP/IP stack will not join the sysplex group and will not process sysplex definitions within the profile (VIPADYNAMIC and IPCONFIG/IPCONFIG6 DYNAMICXCF statements).

In the message text:

tcpstackname
   The name of the TCP/IP stack.

System action
TCP/IP continues.

Operator response
If you want the TCP/IP stack to immediately join the sysplex group, issue the VARY TCPIP,,SYSPLEX,JOINGROUP command. See the information about the VARY TCPIP,,SYSPLEX command in z/OS Communications Server: IP System Administrator's Commands for information about the command and the JOINGROUP parameter.

System programmer response
If you want the stack to join the sysplex group when the stack is started, stop the stack, remove NOJOIN from the GLOBALCONFIG statement in the TCP/IP profile and restart the stack.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFDLM

Routing code
10

Descriptor code
12

Automation
Not applicable.
Example

EZD1928E TCPCS WILL NOT JOIN THE SYSPLEX - GLOBALCONFIG SYSPLEXMONITOR NOJOIN IS Configured

EZD1929I tcpstackname SECURITY DOMAIN NAME NOT DEFINED

Explanation
A SIOCSPARTNERINFO or SIOCGPARTNERINFO ioctl was issued for the TCP/IP stack and failed because a security domain name for the EZBDOMAIN profile was not defined in the SERVAUTH class.

In the message text:

\textit{tcpstackname}

The name of the TCP/IP stack.

System action
TCP/IP continues.

Operator response
Not applicable.

System programmer response
To retrieve your partner security credentials within the sysplex or subplex over a trusted TCP/IP connection, set up a security domain name. See the information about steps for retrieving partner security credentials in \textit{z/OS Communications Server: IP Programmer's Guide and Reference}.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
ezbtcfio

Routing code
2

Descriptor code
12

Automation
Not applicable.

Example

EZD1929I TCPCS SECURITY DOMAIN NAME NOT DEFINED
EZD1930I  HOT STANDBY SERVER FOR destipaddr PORT portnum AT tcpstackname ON mvsname IS INACTIVE - reason

Explanation
The Sysplex Distributor server is switching from the active server because of the reason specified.
In the message text:

destipaddr
The distributed DVIPA.

portnum
The distributed port.

tcpstackname
The name of the TCP/IP stack.

mvsname
The name of the MVS system where the TCP/IP job is running.

reason
The cause of the switch. Possible values are:

THE TARGET SERVER IS NOT READY
The target server has a ready count of 0.

THE PATH TO THE TARGET SERVER IS INACTIVE
The datapath to the XCF target is inactive.

AUTOSWITCHBACK TO THE PREFERRED SERVER OCCURRED
The preferred server was previously either not ready, or the datapath was inactive and the sysplex distribution was switched to a backup server. The preferred server is now ready, or the datapath is active, so the sysplex distribution has switched back to the preferred server.

THE TSR IS 0%
The Target Server Responsiveness of the server is 0%.

THE ABNORMAL TERMINATION LIMIT OF 1000 HAS BEEN REACHED
The target server reported that out of 1000 transactions, all of them terminated abnormally.

THE SERVER HEALTH IS 0%
The target server reported a health of 0%.

System action
TCP/IP continues.

Operator response
Not applicable.

System programmer response
Fix the problem after analyzing the message. If the server is not ready, verify that the target server is in LISTEN state. If you cannot fix the problem, then contact your IBM support center with the TCP/IP profile, system log and a dump.

User response
Not applicable.

Problem determination
Not applicable
**Source**
z/OS Communications Server TCP/IP: Configuration & Initialization

**Module**
EZBXFWLM

**Routing code**
10

**Descriptor code**
12

**Automation**
No

**Example**

```
EZD1930I HOT STANDBY SERVER FOR 10.61.0.1 PORT 6000 AT TCPCS ON VIC018 IS INACTIVE -
THE TARGET SERVER IS NOT READY
```

**EZD1941I**  DCAS DEBUG LEVEL SET TO **debug_level**

**Explanation**
This message displays the active DCAS server debug level. This message is issued when a MODIFY command is issued to the DCAS server to change the debug level.

In the message text:

**debug_level**
The current debug level. See the information about the MODIFY command--DCAS in z/OS Communications Server: IP System Administrator's Commands for more information.

**System action**
DCAS continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP
Unsupoorted DCAS Modify Command (EZD1942I)

**Explanation**
A MODIFY command was issued to DCAS, but DCAS did not recognize the parameter that was entered on the MODIFY command.

**System action**
The request is ignored.

**Operator response**
Correct the MODIFY command and reissue the command. See the information about the MODIFY command--DCAS in z/OS Communications Server: IP System Administrator's Commands for more information.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP
Descriptor code
*

Automation
This message goes to the console as a response to a MODIFY command.

Example

<table>
<thead>
<tr>
<th>EZD1942I UNSUPPORTED DCAS MODIFY COMMAND</th>
</tr>
</thead>
</table>

EZD1943I  UNSUPPORTED DCAS DEBUG LEVEL debug_level

Explanation
A MODIFY command was issued to DCAS to change the debug level. The debug level provided is unsupported.

In the message text:

**debug_level**
The debug level that is not supported.

System action
DCAS continues.

Operator response
Specify a supported debug level and reissue the MODIFY command. See the information about the MODIFY command--DCAS in z/OS Communications Server: IP System Administrator's Commands for more information.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP

Module
DCASSTOP

Routing code
*

Descriptor code
*
Automation
This message goes to the console as a response to a MODIFY command.

Example

EVD1943I UNSUPPORTED DCAS DEBUG LEVEL 9

EZD1944I Unable to open message catalog cat : errno (description) errnojr errnojr - Default messages will be used

Explanation
An attempt was made to open the certbundle command message catalog in the message catalog directory, but the catalog could not be opened for the specified reason. The default location for the message catalog is set by the NLS_PATH environment variable.

In the message text:
- **cat**: The name of the catalog that the certbundle command attempted to open.
- **errno**: The z/OS UNIX System Services return code. These return codes are listed and described in the return codes (ernos) information in z/OS UNIX System Services Messages and Codes.
- **description**: Describes the meaning of the errno value.
- **errnojr**: The hexadecimal z/OS UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
The certbundle command processing continues. Default messages will be used.

Operator response
If the default messages are acceptable, no action is necessary. Otherwise, contact the system programmer to correct the indicated error.

System programmer response
If you want to use the certbundle message catalog, correct the indicated error. If the default messages are acceptable, no action is necessary.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
main.cpp
Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1944I Unable to open message catalog certbundlemsg.cat : errno 113 ( EDC5113I Bad file
descriptor. )
   errnojr 0xC90F0003 - Default messages will be used

EZD1945I Unknown option opt

Explanation
The certbundle command detected an unknown option while parsing the command line.
In the message text:

```
opt
```
   The option that is unknown.

System action
The certbundle command processing ends.

Operator response
See the information about the certbundle command in z/OS Communications Server: IP System
Administrator's Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information
about the certbundle command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
main.cpp

Routing code
Not applicable.
Descriptor code
Not applicable.

Automation
Not applicable.

Example

| EZD1945I Unknown option -h |

EZD1946I Option opt is missing required data

Explanation
The specified certbundle command option requires a value, but no value was specified on the command.

In the message text:

opt
The option that is missing data.

System action
The certbundle command processing ends.

Operator response
Correct and reissue the command. See the information about the certbundle command in z/OS Communications Server: IP System Administrator's Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information about the certbundle command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
main.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.
Example

EZD1946I Option -i is missing required data

EZD1947I Unexpected characters found on line line_number - characters

Explanation
The certbundle command encountered unexpected characters while parsing the options file.

In the message text:

- **line_number**: The line number in the options file on which the unexpected characters were found.
- **characters**: The unexpected characters that were found in the options file

System action
The certbundle command processing ends.

Operator response
Contact the system programmer.

System programmer response
Correct the options file syntax and reissue the command. See the information about the certbundle command in z/OS Communications Server: IP System Administrator's Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information about the certbundle command syntax and options.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CBCParser.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.
EZD1948I  Could not open file_type (file_name) - description

Explanation
The certbundle command was unable to open the specified file.
In the message text:

file_type
The type of file that the command failed to open. Some options are the CertBundleOptions, CRLFile, or BundleFile.

file_name
The name of the file that the command failed to open.

description
A description of the system error encountered by the command.

System action
The certbundle command processing ends.

Operator response
Ensure that the file name exists at the specified location and that the certbundle command has permission to read the file.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CBCParser.cpp CertBundle.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.
Example

EZD1948I Could not open CertBundleOptions ( /tmp/options_file ) - EDC5113I Bad file descriptor.

EZD1949I CertBundleOptions statement beginning on line line_number references a self-signed certificate ( label )

Explanation

The certbundle command parsed a CertBundleOptions statement that references a self-signed certificate. Self-signed certificates do not need to be placed in a certificate bundle.

In the message text:

*line_number*
  The line number in the options file where the CertBundleOptions statement begins.

*label*
  The certificate label of the self-signed certificate that was referenced.

System action

The self-signed certificate will be included in the bundle and certbundle command processing continues.

Operator response

Contact the system programmer.

System programmer response

Determine if the self-signed certificate is required in the certificate bundle. If it is not required, remove the CertificateLabel or CertificateChain parameter that references the self-signed certificate and reissue the command.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server: z/OS UNIX certbundle command

Module

CertBundle.cpp

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.
Example

EZD1949I CertBundleOptions statement beginning on line 2 references a self-signed certificate (FVT Root1 CA)

EZD1950I Primary option not specified

Explanation
The certbundle command requires a primary option but none was provided.

System action
The certbundle command processing ends.

Operator response
Correct and reissue the command. See the information about the certbundle command in z/OS Communications Server: IP System Administrator’s Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information about the certbundle command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
main.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

EZD1950I Primary option not specified

EZD1951I file_type file not specified
Explanation
A required input file was not specified.

In the message text:

*file_type*
   The type of required input file.

System action
The `certbundle` command processing ends.

Operator response
Correct and reissue the command. See the information about the `certbundle` command in z/OS Communications Server: IP System Administrator's Commands or issue the `man certbundle` command in a z/OS UNIX shell to obtain information about the `certbundle` command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
main.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

| EZD1951I | CertBundleOptions file not specified |

| EZD1952I | *keyword* on line *line_number* is not a recognized parameter or statement |

Explanation
The `certbundle` command encountered an unrecognized parameter or statement while parsing the options file.
**In the message text:**

*keyword*
The unrecognized parameter or statement.

*line_number*
The line number in the options file that contains the unrecognized keyword.

**System action**
The `certbundle` command processing ends.

**Operator response**
Correct the options file syntax and reissue the command. See the information about the `certbundle` command in z/OS Communications Server: IP System Administrator's Commands or issue the `certbundle` command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX certbundle command

**Module**
CBCParser.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**

```
EZD1952I CertLabel on line 22 is not a recognized parameter or statement

EZD1953I Incorrect opt option value value is ignored
```

**Explanation**
An incorrect option value was specified and ignored.

In the message text:
opt
  The certbundle command option that was specified.

value
  The value that was specified for the certbundle command option.

System action
The certbundle command continues, using any specified valid values or any default values.

Operator response
Specify an option value that is in the accepted value range and issue the certbundle command again. See the information about the certbundle command in z/OS Communications Server: IP System Administrator's Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information about the certbundle command syntax and options.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
main.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

EZD1953I Incorrect -d option value 4 is ignored

EZD1954I CertBundleOptions statement beginning on line line_number will not include self-signed certificate (label) in the bundle

Explanation
The certbundle command parsed a CertBundleOptions statement that referenced a self-signed certificate. This self-signed certificate was found while parsing the certificate hierarchy specified by the CertificateChain
parameter. Certificates that are not self-signed will be included in the bundle, but the self-signed certificate will not be included in the bundle.

In the message text:

**line_number**
- The line number in the options file where the CertBundleOptions statement begins.

**label**
- The certificate label of the self-signed certificate that was referenced.

**System action**
The `certbundle` command processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX certbundle command

**Module**
CertBundle.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**

<table>
<thead>
<tr>
<th>EZD19541 Warning - CertBundleOptions statement beginning on line 2 will not include self-signed certificate ( FVT Root1 CA ) in the bundle</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD1955I CertBundleOptions statement beginning on line <strong>line_number</strong> is missing <strong>keyword</strong> parameter</td>
</tr>
</tbody>
</table>

**Explanation**
The `certbundle` command found that a parameter was missing from a CertBundleOptions statement in the options file.
In the message text:

**line_number**
The line number in the options file where the CertBundleOptions statement begins.

**keyword**
The missing parameter.

**System action**
The certbundle command processing ends.

**Operator response**
Contact the system programmer.

**System programmer response**
Correct the options file syntax and reissue the command. See the information about the certbundle command in z/OS Communications Server: IP System Administrator's Commands or issue the certbundle command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX certbundle command

**Module**
CertBundle.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**

```
EZD1955I  CertBundleOptions statement beginning on line 2 is missing BundleFile  parameter
```

```
EZD1956I  An expected value is missing on line line_number
```

**Explanation**
The certbundle command encountered a missing value for a parameter specified in the options file.

In the message text:
**line_number**
The line number in the options file where the parameter that is missing a value was found.

**System action**
The `certbundle` command processing ends.

**Operator response**
Contact the system programmer.

**System programmer response**
Correct the options file syntax and reissue the command. See the information about the `certbundle` command in z/OS Communications Server: IP System Administrator's Commands or issue the `man certbundle` command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX certbundle command

**Module**
CBCParser.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**

```
EZD1956I Expected value missing on line 15
EZD1957I type brace ( brace ) expected, but not found
```

**Explanation**
The `certbundle` command encountered an error while parsing the options file. A brace was expected but not found.

In the message text:

- **type**
  The type of brace that was expected. Opening braces ({{}) and closing braces (}}) are expected throughout the options file.
**brace**

The literal character that was expected within the options file.

**System action**

The `certbundle` command processing ends.

**Operator response**

Contact the system programmer.

**System programmer response**

Correct the options file syntax and reissue the command. See the information about the `certbundle` command in z/OS Communications Server: IP System Administrator's Commands or issue the `man certbundle` command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server: z/OS UNIX certbundle command

**Module**

CBCParser.cpp

**Routing code**

Not applicable.

**Descriptor code**

Not applicable.

**Automation**

Not applicable.

**Example**

EZD1957I Closing brace (}) expected, but not found

**EZD1958I type brace (brace) found, but not expected**

**Explanation**

The `certbundle` command encountered an error while parsing the options file. A brace was found, but not expected.

In the message text:

**type**

The type of brace that was found. Opening braces (({) and closing braces (}) may be found throughout the options file.
brace
The literal character that was found within the options file.

System action
The certbundle command processing ends.

Operator response
Contact the system programmer.

System programmer response
Correct the options file syntax and reissue the command. See the information about the certbundle command in z/OS Communications Server: IP System Administrator's Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CBCParser.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>EZD1958I</th>
<th>Closing brace ( { } ) found, but not expected</th>
</tr>
</thead>
</table>

| EZD1961I | file_type ( file ) created successfully |

Explanation
The certbundle command successfully created an output file.

In the message text:

file_type
The type of output file that was created. The most common type of output file is a bundle file.

file
The absolute path name of the output file that was created.
System action
The certbundle command processing continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CertBundle.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1961I BundleFile ( /u/user1/Bundles/SalesDeptBundle.der ) created successfully

EZD1962I  Failed to open KeyRing keyring - description

Explanation
The certbundle command failed to open a KeyRing that was specified in the options file.

In the message text:

keyring
   The name of the keyring that the certbundle command failed to open.

description
   The GSK description of the error that is preventing the command from opening the keyring.

System action
The certbundle command processing ends.
Operator response
Contact the system programmer.

System programmer response
Ensure that the keyring exists and that the user ID under which the certbundle command is running has the appropriate SAF permission to open the keyring. Then reissue the command.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CertBundle.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>EZD1962I Failed to open KeyRing NSSD/NoAccess - Access denied</th>
</tr>
</thead>
</table>

EZD1963I Certificate with label ( label ) not found

Explanation
The certbundle command was unable to locate the certificate with the specified label on the keyring.

In the message text:

label
The label of the certificate that was not found on the keyring.

System action
The certbundle command processing ends.

Operator response
Contact the system programmer.
**System programmer response**

Ensure that the certificate with the specified label exists on the keyring and reissue the command. The keyring is specified in the options file.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server: z/OS UNIX certbundle command

**Module**

CertBundle.cpp

**Routing code**

Not applicable.

**Descriptor code**

Not applicable.

**Automation**

Not applicable.

**Example**

<table>
<thead>
<tr>
<th>EZD1963I Certificate with label ( SomeCertLabel ) not found</th>
</tr>
</thead>
</table>

**EZD1964I**  **Error creating bundle file** *(file_name)* - *description*

**Explanation**

The `certbundle` command encountered an error while it was creating the bundle file.

In the message text:

*file_name*  
The name of the bundle file that the `certbundle` command failed to create.

*description*  
A description of the system error that caused the failure.

**System action**

The `certbundle` command processing ends.

**Operator response**

Contact the system programmer.
System programmer response
Ensure that the path name of the bundle file is specified correctly, that the path exists, that the file system is mounted in read/write mode, and that the user ID under which the command is running has permission to write to the file system. Then reissue the command.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CertBundle.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1964I Error creating BundleFile (/usr/lpp/BundleFile) (EDC5141I Read-only file system.)
EZD1965I CertBundleOptions statement beginning on line line_number does not support both CertificateLabel and CertificateChain parameters

Explanation
The certbundle command encountered an error while it was parsing the options file. A CertBundleOptions statement was found that contains both CertificateLabel and CertificateChain parameters. A single CertBundleOptions statement can support only one of these parameters.

In the message text:

line_number
The line number on which the CertBundleOptions statement started.

System action
The certbundle command processing ends.

Operator response
Contact the system programmer.
System programmer response
Correct the options file syntax and reissue the command. See the information about the certbundle command in z/OS Communications Server: IP System Administrator's Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CertBundle.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

EZD1965I CertBundleOptions statement beginning on line 6 does not support both CertificateLabel and CertificateChain parameters

EZD1966I parameter parameter unexpected on line new_line because the parameter is already set to value on line old_line

Explanation
The certbundle command found more than one definition for the specified parameter while parsing the options file. This parameter can be defined only once.

In the message text:

parameter
The parameter that was previously defined.

value
The value to which the parameter was previously defined.

old_line
The line on which the parameter was originally defined.

new_line
The line on which the parameter value was provided in error.

System action
The certbundle command processing ends.
Operator response
Contact the system programmer.

System programmer response
Correct the options file syntax and reissue the command. See the information about the certbundle command in z/OS Communications Server: IP System Administrator's Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CertBundle.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>Value that you enter</th>
<th>Value that is interpreted</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;EndEntity&quot;&quot;Bob&quot;&quot;Cert&quot;</td>
<td>EndEntity&quot;Bob&quot;Cert</td>
<td>The value &quot;Bob&quot; includes double quotation marks</td>
</tr>
<tr>
<td>&quot;EndEntityBob&quot;&quot;Cert&quot;&quot;</td>
<td>EndEntityBob&quot;Cert&quot;</td>
<td>The value &quot;Cert&quot; includes double quotation marks</td>
</tr>
</tbody>
</table>

EZD1967I Unbalanced or extra quote found on line line_number

Explanation
The certbundle command encountered an error as the result of incorrectly placed quotation marks while it was parsing the options file. If a parameter contains a value that itself contains double quotation marks, that value must be enclosed in two sets of double quotation marks, and the entire parameter must be enclosed in another set of double quotation marks. Only one value in a parameter value can contain quotation marks.

Examples:

<table>
<thead>
<tr>
<th>Value that you enter</th>
<th>Value that is interpreted</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;EndEntity&quot;&quot;Bob&quot;&quot;Cert&quot;</td>
<td>EndEntity&quot;Bob&quot;Cert</td>
<td>The value &quot;Bob&quot; includes double quotation marks</td>
</tr>
<tr>
<td>&quot;EndEntityBob&quot;&quot;Cert&quot;&quot;</td>
<td>EndEntityBob&quot;Cert&quot;</td>
<td>The value &quot;Cert&quot; includes double quotation marks</td>
</tr>
</tbody>
</table>

System action
The certbundle command processing ends.

1308 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Operator response
Contact the system programmer.

System programmer response
Correct the options file syntax and reissue the command. See the information about the certbundle command in z/OS Communications Server: IP System Administrator's Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CBCParser.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1967I Unbalanced or extra quote found on line 14

EZD1968I CertBundleOptions statement beginning on line line_number references a self-signed certificate (label) but contains no CRLFile statement - bundle creation might be unnecessary

Explanation
The certbundle command parsed a CertBundleOptions statement that referenced a self-signed certificate. The same CertBundleOptions statement did not contain a CRLFile statement. The self-signed certificate will be added to the certificate bundle but the creation of a bundle file might be unnecessary.

In the message text:

line_number
The line number in the options file on which the CertBundleOptions statement begins.

label
The certificate label of the self-signed certificate that was referenced.

System action
The certbundle command processing continues.
Operator response
Contact the system programmer.

System programmer response
Determine if the self-signed certificate is required in the certificate bundle. If it is not required, remove the CertificateLabel or CertificateChain parameter that references the self-signed certificate and reissue the command.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CertBundle.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1968I CertBundleOptions statement beginning on line 6 references a self-signed certificate (FVT Root1 CA) but contains no CRLFile statement - bundle creation might be unnecessary

EZD1969I Unexpected parameter parameter found on line line_number

Explanation
The certbundle command encountered an error while parsing the options file. An unexpected parameter was found. Parameters must be contained within a valid statement block.

In the message text:
parameter
The unexpected parameter that was found.

line_number
The line number on which the parameter was found.

System action
The certbundle command processing ends.
Operator response
Contact the system programmer.

System programmer response
Correct the options file syntax and reissue the command. See the information about the `certbundle` command in z/OS Communications Server: IP System Administrator's Commands or issue the `man certbundle` command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
CBCParser.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example

```
EZD1969I Unexpected parameter BundleFile found on line 16
```

```
EZD1970I HOT STANDBY SERVER FOR destipaddr PORT portnum AT tcpstackname ON mvsname IS ACTIVE
```

Explanation
The Sysplex Distributor switched to the specified hot standby server.

In the message text:

- **destipaddr**
  - The distributed DVIPA.

- **portnum**
  - The distributed port.

- **tcpstackname**
  - The name of the TCP/IP stack.

- **mvsname**
  - The name of the MVS system where the TCP/IP job is running.
System action
TCP/IP continues.

Operator response
Not applicable.

System programmer response
Not applicable.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBXFWLM

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD1970I HOT STANDBY SERVER FOR 10.61.0.2 PORT 6000 AT TCPCS ON VIC018 IS  ACTIVE

EZD1971I  SYSPLEX DISTRIBUTOR CONTROL CONNECTION TO DESTINATION IP ADDRESS ip_addr FAILED WITH RETURN CODE errno REASON CODE errnojr

Explanation
The control connection to the non-z/OS target could not be established. The non-z/OS target cannot be used for distributed connections. The control connection can fail if no route is available to the target, or if there is a problem with the sysplex distributor agent server on the target.

In the message text:

ip_addr
The IP address of a distributed non-z/OS target.

errno
A UNIX System Services return code from a failing Connect() call. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes.
**errnojr**
The hexadecimal UNIX System Services reason code from a failing Connect() call. The format of the 4-byte reason code is explained in the introduction to the reason codes (errnojrs) information in the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
TCP/IP continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Fix the problem after analyzing the return code and the reason code. For example, if the return code is 00000468 (ECONNREFUSED), verify that the sysplex distributor agent server is in LISTEN state and that it is accepting new connections. After you fix the problem, the sysplex distributor will establish the control connection because the sysplex distributor periodically tries the connection attempt again.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: Sysplex Distributor

**Module**
EZBXFSUB

**Routing code**
2,8

**Descriptor code**
12

**Automation**
This message is written to the system console. This message is a good candidate for automation. Automation can allow you to monitor the status of sysplex distributor control connections to non-z/OS targets.

**Example**

```
EZD1971I SYSPLEX DISTRIBUTOR CONTROL CONNECTION TO DESTINATION IP ADDRESS 9.42.105.53 FAILED WITH RETURN CODE 00000468 REASON CODE 00000000
```

```
EZD1972I VIPADISTRIBUTE keyword IS NOT VALID BECAUSE THE CURRENT DISTRIBUTION METHOD IS NOT HOTSTANDBY
```
**Explanation**

The DISTMETHOD parameter was not specified on the VIPADISTRIBUTE statement. This keyword is valid only when the DISTMETHOD is HOTSTANDBY. Either the DISTMETHOD defaulted to BASEWLM, or a previous VIPADISTRIBUTE statement for this DVIPA and port set DISTMETHOD to a value other than HOTSTANDBY.

In the message text:

*keyword*

The keyword that is not valid.

**System action**

TCP/IP continues. The VIPADISTRIBUTE statement is rejected.

**Operator response**

Not applicable.

**System programmer response**

Change the VIPADISTRIBUTE DISTMETHOD to HOTSTANDBY or remove the keyword. Then issue a VARY TCPIP,OBEYFILE command that specifies a data set file that contains the entire VIPADYNAMIC block. See the information about the VIPADYNAMIC statement in z/OS Communications Server: IP Configuration Reference for more information.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZBXFDV2, EZBX6DV2

**Routing code**

10

**Descriptor code**

12

**Automation**

Not applicable.

**Example**

```
EZD1972I VIPADISTRIBUTE AUTOSWITCHBACK IS NOT VALID BECAUSE THE CURRENT DISTRIBUTION METHOD IS NOT HOTSTANDBY

EZD1973E MULTIPLE tcpstackname NONRECOVERABLE ERRORS ARE ADVERSELY AFFECTING SYSPLEX PROCESSING
```
Explanation

Five TCP/IP abends occurred within one minute.

In the message text:

```
tcpstackname
```

The name of the TCP/IP stack.

System action

TCP/IP continues.

- If the GLOBALCONFIG SYSPLEXMONITOR RECOVERY option is active and this stack is not the only member of its TCP/IP sysplex group, the following RECOVERY actions occur:
  - This stack leaves the TCP/IP sysplex group.
  - This stack no longer participates in sysplex distribution (as a distributor or target) or acts as an owner or a backup for DVIPAs. All DVIPAs defined on this stack are deactivated; however, the DVIPA definitions are saved.
  - When the stack leaves the TCP/IP sysplex group, this operator message is deleted.
- If the GLOBALCONFIG SYSPLEXMONITOR NORECOVERY option is active, no action is taken.

See the information about sysplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide for more information.

See the GLOBALCONFIG configuration statement in z/OS Communications Server: IP Configuration Reference for more information about the SYSPLEXMONITOR parameters.

Operator response

Save the documentation that was taken when the problem occurred and contact the system programmer.

If the NORECOVERY option is active, no further actions are needed.

If the RECOVERY option is active, then even if the GLOBALCONFIG SYSPLEXMONITOR AUTOREJOIN option is active, the stack will not automatically rejoin the TCP/IP sysplex group. Message EZZ9676E will be displayed if the TCP/IP stack successfully deactivates all DVIPAs and leaves the TCP/IP sysplex group. After message EZZ9676E is displayed, issue the VARY TCPIP,,SYSPLEX,JOINGROUP command to cause the DVIPA definitions to be processed and to cause the stack to rejoin the TCP/IP sysplex group.

System programmer response

Contact IBM software support services with the TCP/IP profile, system log, and dump.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP

Module

EZBXFPDC
Routing code
2,8

Descriptor code
2

Automation
Not applicable.

Example
MULTIPLE TCPCS NONRECOVERABLE ERRORS ARE ADVERSELY AFFECTING SYSPLEX PROCESSING

EZD1974E tcpstackname CSM HAS BEEN CONSTRAINED FOR AT LEAST timevalue SECONDS

Explanation
Sysplex problem detection determined that storage managed by the communications storage manager (CSM) has been constrained for multiple monitoring intervals.

In the message text:

tcpstackname
The name of the TCP/IP stack.

timevalue
The number of seconds that syplex problem detection has determined that CSM has been constrained.

System action
TCP/IP continues.

If the GLOBALCONFIG SYSPLEXMONITOR RECOVERY option is active, and this stack is not the only member of the TCP/IP sysplex group, the following RECOVERY actions will occur:

- This stack will leave the sysplex group.
- This stack will no longer participate in sysplex distribution (as a distributor or target) or act as an owner or a backup for DVIPAs. All DVIPAs defined on this stack will be deactivated; however, the DVIPA definitions will be saved.
- If the problem is corrected, this operator message will be deleted. If the GLOBALCONFIG SYSPLEXMONITOR AUTOREJOIN option is active, the stack will process the DVIPA definitions and rejoin the TCP/IP sysplex group.

If the GLOBALCONFIG SYSPLEXMONITOR NORECOVERY option is active, no action will be taken. Monitoring will continue. This operator message will be deleted when the CSM constraint is corrected.

See syplex problem detection and recovery in z/OS Communications Server: IP Configuration Guide for more information.

Operator response
Save the TCP/IP profile and system log. If a dump was not created, take a dump of the TCP/IP address space and dataspaces, and the CSM dataspace.

System programmer response
Messages were issued before this message to report that CSM storage is constrained. Those messages identify the type of CSM storage that is constrained. See the documentation for those messages for the actions that you must take to resolve the storage constraint.
If the CSM storage problem cannot be corrected, contact your IBM support center with the documentation that was obtained when the problem occurred.

If the CSM storage problem can be corrected:

- If RECOVERY is being used, enable the stack to rejoin the sysplex group. Message EZZ9676E is issued after the process of leaving the sysplex group has successfully completed. After this message is issued, reapply the sysplex profile definitions by issuing VARY OBEY. This will cause the stack to rejoin the sysplex group.
- If NORECOVERY is being used, no further actions are needed.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server TCP/IP: Configuration & Initialization

**Module**

EZBXFPDM

**Routing code**

2, 8

**Descriptor code**

2

**Automation**

Not applicable.

**Example**

```
EZD1974E TCPCS1 CSM HAS BEEN CONSTRAINED FOR AT LEAST 90 SECONDS
```

**EZD1975E** UNABLE TO CLAIM A LIST IN THE *structure_name* STRUCTURE

**Explanation**

An attempt to claim a list in the identified EZBDVIPA coupling facility structure failed because a list was not available.

In a sysplex, an EZBDVIPA coupling facility structure is required to support Sysplex-wide Security Associations (SWSA). Tunnel sequence numbers are stored in the EZBDVIPA structure to enable sysplex distribution. Tunnel data is stored in the EZBDVIPA structure to enable DVIPA takeover.

Depending on the number of DVIPAs and IPSec tunnels in use, TCP/IP can exhaust the number of lists defined in the EZBDVIPA structure. Once all lists have been claimed, subsequent attempts to claim a list fail causing data traffic over the affected tunnel to fail if the traffic is distributed. Also, the affected tunnel cannot be recovered after a DVIPA takeover.

In the message text

*structure_name*

The name of the EZBDVIPA coupling facility structure for which the claim list failed.
Guideline: If subplexing is not in use, the name of the structure is EZBDVIPA. If you use subplexing within your sysplex, the name is in the form EZBDVIPAvvtt, where vv is the VTAM group ID and tt is the TCP/IP group ID.

The message stays on the console until new tunnels are successfully added to the structure.

System action
TCP/IP continues. Data traffic over the affected tunnel fails if the traffic is distributed. The affected tunnel can not be recovered after a DVIPA takeover.

Operator response
Contact the system programmer.

System programmer response
Issue D NET,STATS,TYPE=VTAM,STRNAME=structure_name on each VTAM in the sysplex to determine the number of lists in the coupling facility structure.

If message IST1189I appears in the DISPLAY STATS output, this indicates that VTAM might not have access to all the lists. For additional information on the DISPLAY STATS output, see the description of IST1189I under the first message in the display, IST1370I. See Modifying the number of lists in z/OS Communications Server: SNA Network Implementation Guide for instructions on how to adjust the number of lists that VTAM can access in the EZBDVIPA structure.

If your configuration has a large number of DVIPAs and you expect a large number of tunnels between endpoints, see Modifying the number of lists in z/OS Communications Server: SNA Network Implementation Guide for instructions on how to increase the number of lists for the EZBDVIPA structure.

Otherwise, evaluate your policy definitions to identify the reason for the unexpectedly large number of tunnels.

User response
Not applicable.

Problem determination
See the System Programmer Response.

Source
z/OS Communications Server TCP/IP

Module
EZBXFCFS, EZBXFCFP, EZBXFVET

Routing code
2,8

Descriptor code
12

Automation
Not applicable.
EZD1980I CertBundleOptions statement beginning on line line_number is missing a closing brace ()

Explanation
The certbundle command encountered an error while it was parsing the options file. A CertBundleOptions statement was missing a closing brace (}).

In the message text:

line_number
The line number in the options file on which the CertBundleOptions statement begins.

System action
The certbundle command processing ends.

Operator response
Contact the system programmer.

System programmer response
Correct the options file syntax and reissue the command. See the information about the certbundle command in z/OS Communications Server: IP System Administrator's Commands or issue the man certbundle command in a z/OS UNIX shell to obtain information about the certbundle options file syntax.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command
EZD1981I Storage allocation failed

Explanation
The certbundle command failed to allocate memory for internal processing.

System action
The certbundle command processing ends.

Operator response
The error might be transient; reissue the request. If the error persists, contact the system programmer.

System programmer response
Trace or log entries might provide more information about the error. Ensure that there is enough memory available on the system. See the information about diagnosing storage abends and storage growth in z/OS Communications Server: IP Diagnosis Guide for more information about storage problems.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server: z/OS UNIX certbundle command

Module
main.cpp CBCParser.cpp CertBundle.cpp

Routing code
Not applicable.

Descriptor code
Not applicable.

Automation
Not applicable.

Example
EZD1981I Storage allocation failed

EZD1982I GSK function call failed with status code error - description

Explanation
A call to the System SSL Certificate Management Services (CMS) API returned an error.
In the message text:

**function**
The API call that failed. See the System SSL CMS API in z/OS Cryptographic Services System SSL Programming for more information.

**error**
The hexadecimal CMS status code. See CMS status codes in z/OS Cryptographic Services System SSL Programming for more information.

**description**
Describes the meaning of the error.

**System action**
The certbundle command processing ends.

**Operator response**
Contact the system programmer.

**System programmer response**
Use error and the description provided to fix the error.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX certbundle command

**Module**
CertBundle.cpp

**Routing code**
Not applicable.

**Descriptor code**
Not applicable.

**Automation**
Not applicable.

**Example**

```
EZD1982I GSK gsk_decode_crl call failed with status code 0x014ce00e - Data type is not correct
```
Chapter 13. EZD2xxxx messages

EZD2011I THE VARY TCPIP,,DROP COMMAND WAS IGNORED BECAUSE THE COMMAND PARAMETERS DID NOT MATCH A LISTENING APPLICATION

Explanation
A VARY TCPIP,,DROP command was issued with parameters used to select a listening application. The command was ignored because no listening application was found that matched the parameters that were specified on the command.

System action
TCP/IP continues.

Operator response
Change the parameters specified on the VARY TCPIP,,DROP command so that they match an active listening application, and re-issue the command.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBTCICT

Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example
Not applicable.

EZD2012I THE VARY TCPIP,,DROP COMMAND WAS REJECTED BECAUSE MORE THAN ONE LISTENING APPLICATION WAS FOUND THAT MATCHED THE COMMAND PARAMETERS
Explanation
A VARY TCPIP,,DROP command was issued. The command was rejected because there was more than one listening application that matched the parameters specified on the command.

System action
TCP/IP continues.

Operator response
Re-issue the command with the JOBNAME parameter and possibly the ASID parameter to identify a unique listening application.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: Configuration & Initialization

Module
EZBTCICT

Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example
Not applicable.

**EZD2013I**  
numconns CONNECTIONS WERE SUCCESSFULLY DROPPED

Explanation
A VARY TCPIP,,DROP command was issued. The command completed successfully. This message displays the number of connections that were dropped.

In the message text:

**numconns**  
The number of connections that were successfully dropped. If the **numconns** value is 0, then no connections associated with the server that matched the input parameters were found.
System action
TCP/IP continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBTCICT

Routing code
2,8

Descriptor code
12

Automation
Not applicable.

Example
EZD2013I 10 CONNECTIONS WERE SUCCESSFULLY DROPPED

EZD2014I  Real-time application-controlled TCP/IP trace NMI has been disabled – reason code rsncode

Explanation
The real-time application-controlled TCP/IP trace network management interface (NMI) has been disabled because of an error. The rsncode value provides the reason for the error. See Real-time application-controlled TCP/IP trace NMI in z/OS Communications Server: IP Programmer’s Guide and Reference for more information about this NMI.

In the message text:

rsncode
A code that explains the error. Possible values are:

1
Unable to obtain the TCP/IP address space private storage for internal control blocks

2
Unable to obtain the 64-bit common storage for internal control blocks
Internal error

System action
TCP/IP processing continues, but applications will not be able to use the real-time application-controlled TCP/IP trace NMI.

Operator response
Save the system log for problem determination and contact the system programmer.

System programmer response
When the NMI has been disabled, the TCP/IP stack must be recycled to enable it. Use the rsncode value to determine what action to take before recycling the TCP/IP stack:

1. Increase the size of the TCP/IP address space to ensure that enough private storage is available for use by the NMI. If you continue to receive this message with a rsncode value of 1, obtain a dump of the TCP/IP address space and examine the private storage usage. Under IPCS, you can use the z/OS Communications Server TCPIPCS MAP command to examine information from the dump about storage use by the TCP/IP stack. See TCPIPCS MAP in z/OS Communications Server: IP Diagnosis Guide for more information about this command.

2. The amount of 64-bit common storage is controlled by the HVCOMMON parameter in member IEASYSxx of PARMLIB. For more information about this parameter, see the z/OS MVS Initialization and Tuning Reference. You can use the MVS console command D VIRTSTOR,HVCOMMON to display the summary information about the current use of the HVCOMMON storage on the MVS systems. For more information about this command, see the z/OS MVS System Commands. Review the value specified on the HVCOMMON parameter to determine whether it should be increased. If you continue to receive this message with a rsncode value of 2, obtain a system dump and examine the 64-bit common storage usage. Under IPCS, you can use the RSMDATA HVCOMMON command to display detailed information about the users of the storage and the ranges in use. For more information about this command, see the z/OS MVS Diagnosis: Reference.

3. Obtain a dump of the TCP/IP address space and contact the IBM support center.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBRCINI, EZBRCOC2

Routing code
10

Descriptor code
12

1326  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
Automation
You can set up the automation for this message, so you will know when the NMI has been disabled.

Example

<table>
<thead>
<tr>
<th>Message ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD2014I</td>
<td>Real-time application-controlled TCP/IP trace NMI has been disabled – reason code 2</td>
</tr>
</tbody>
</table>

**EZD2015I  ICMPv6 WILL IGNORE REDIRECTS DUE TO INTRUSION DETECTION POLICY**

Explanation
Intrusion Detection Services (IDS) policy is active, and the ICMP_REDIRECT attack policy specifies that ICMPv6 redirect packets are to be discarded. All future ICMPv6 redirects will be ignored.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP

Module
EZBIDATK

Routing code
2, 8

Descriptor code
8, 9

Automation
Not applicable.

Example

<table>
<thead>
<tr>
<th>Message ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD2015I</td>
<td>ICMPv6 WILL IGNORE REDIRECTS DUE TO INTRUSION DETECTION POLICY</td>
</tr>
</tbody>
</table>

**EZD2016I  DISPLAY TRACE for stackname**
Explanation
This message indicates the beginning of the output of the `DISPLAY TCPIP,,TRACE` command. It provides the TCP/IP stack name for which the command was started. For a description of the command output, see `DISPLAY TCPIP,,TRACE` command in z/OS Communications Server: IP System Administrator’s Commands.

In the message text:

`stackname`
The name of the TCP/IP stack to which the command was directed.

System action
The `DISPLAY TCPIP,,TRACE` command continues.

Operator response
None.

System programmer response
None.

User response
None.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZACDDTR

Routing code
*

Descriptor code
8, 9

Automation
Not applicable for automation.

Example
Not applicable.

EZD2017I  ICSF services are currently unavailable to the IKE daemon operating in FIPS 140 mode

Explanation
The IKE daemon has been configured for FIPS 140 mode and is initializing. ICSF is not currently active. The IKE daemon requires services from ICSF when configured for FIPS 140 mode.
**System action**
The IKE daemon ends.

**Operator response**
Contact the system programmer.

**System programmer response**
The IKE daemon fails to initialize if it is configured for FIPS 140 mode and ICSF is not active.
If you want the IKE daemon to operate in FIPS 140 mode, start ICSF and then restart the IKE daemon.
If you do not want the IKE daemon to operate in FIPS 140 mode, specify "FIPS140 No" in the IKE configuration file and restart the IKE daemon.

**User response**
None.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
ike_config.cpp

**Routing code**
2, 8

**Descriptor code**
12

**Automation**
This message is sent to the system console and to syslogd.

**Example**
```
EZD2017I ICSF services are currently unavailable to the IKE daemon operating in FIPS 140 mode
```

**Explanation**
This message is issued as part of a message group. See message EZZ8453I in z/OS Communications Server: IP Messages Volume 4 (EZZ, SNM) for a complete description of the message group.

**System action**
See message EZZ8453I.
EZD2019I  ICSF services are currently available to the IKE daemon

Explanation
The IKE daemon is initializing and ICSF is active.

System action
Processing continues.

Operator response
None.

System programmer response
None.

User response
None.

Problem determination
None.
Source
z/OS Communications Server TCP/IP: IKE daemon

Module
ike_config.cpp

Routing code
2, 8

Descriptor code
12

Automation
This message is sent to the system console and to syslogd.

Example
EZD2019I ICSF services are currently available to the IKE daemon

EZD2020A QDIO ACCELERATOR IS ENABLED ONLY FOR SYSPLEX DISTRIBUTOR BECAUSE OF TCPIP PROFILE FILTER RULES

Explanation
QDIO Accelerator will accelerate only Sysplex Distributor traffic. Routed traffic cannot be accelerated for one of the following reasons:

- You have no IP security filters configured in your TCP/IP profile, or the current IP security filters configured in your TCP/IP profile do not explicitly permit all routed traffic.
- Filter logging is enabled for routed traffic in your TCP/IP profile.

To satisfy your configured IP filters, routed traffic must be processed by the TCP/IP stack, where IP filtering is implemented.

System action
Processing continues with QDIO Accelerator enabled only for sysplex distributor traffic.

Operator response
Contact the system programmer.

System programmer response
No action is required if any of the following conditions are true:

- Your security policy requires some routed traffic to be denied
- Your security policy requires some routed traffic to be protected using IPsec
- Your security policy requires some routed traffic to be subject to filter logging
- You do not want to use QDIO acceleration for your routed traffic

If your security policy allows all routed traffic to be permitted and does not require any routed traffic to be subject to filter logging, you can change the IP security filters configured in your TCP/IP profile so that QDIO Accelerator is enabled for routed traffic. To do this, modify the IPSEC statement in your TCP/IP profile:
1. Ensure that the first IPv4 IPSECRULE statement with a ROUTING specification of ROUTED or EITHER permits all IPv4 addresses, all protocols, and all security classes.

**Tip:** If your rule has a ROUTING specification of EITHER, it applies to both local and routed traffic. If your security policy does not allow you to permit all local traffic, split this rule into two rules, one with a ROUTING specification of ROUTED and one with a ROUTING specification of LOCAL.

2. Ensure that this IPSECRULE statement does not specify LOG to enable filter logging.

For more information, see QDIO Accelerator and IP security in z/OS Communications Server: IP Configuration Guide.

**User response**

Not applicable.

**Problem determination**

None.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZBIPEPR, EZBISPSW, EZBISEPR, EZBISSFT

**Routing code**

10

**Descriptor code**

2

**Automation**

You can automate on this message to detect situations when routed traffic is not being QDIO accelerated because of IP filtering rules.

**Example**

```
EZD2020A  QDIO ACCELERATOR IS ENABLED ONLY FOR SYSPLEX DISTRIBUTOR BECAUSE OF TCPIP PROFILE FILTER RULES

EZD2021A  QDIO ACCELERATOR IS ENABLED ONLY FOR SYSPLEX DISTRIBUTOR BECAUSE OF POLICY FILTER RULES
```

**Explanation**

QDIO Accelerator will accelerate only Sysplex Distributor traffic. Routed traffic cannot be accelerated for one of the following reasons:

- The current IP security filters in your policy configuration do not explicitly permit all routed traffic.
- Filter logging is enabled for routed traffic in your policy configuration.

To satisfy your configured IP filters, routed traffic must be processed by the TCP/IP stack, where IP filtering is implemented.

**System action**

Processing continues with QDIO Accelerator enabled only for sysplex distributor traffic.
Operator response
Contact the system programmer.

System programmer response
No action is required if any of the following conditions are true:

- Your security policy requires some routed traffic to be denied
- Your security policy requires some routed traffic to be protected using IPsec
- Your security policy requires some routed traffic to be subject to filter logging
- You do not want to use QDIO acceleration for your routed traffic

If your security policy allows all routed traffic to be permitted and does not require any routed traffic to be subject to filter logging, you can change the IP security filters in your policy configuration so that QDIO Accelerator is enabled for routed traffic. To do this, modify your policy configuration:

1. If you are using the IBM Configuration Assistant for z/OS Communications Server to configure your IPSec policy:
   a. Ensure that the first connectivity rule that applies to routed IPv4 traffic specifies a topology of filtering only, applies to all IPv4 addresses, and uses a requirement map that maps all IP protocols and all security classes to a security level of Permit.
   
   **Tip:** Your connectivity rule might apply to both local and routed traffic. If your security policy does not allow you to permit all local traffic, split this rule into two rules, one that applies to filtering for routed traffic, and one that applies to filtering for local traffic.
   b. Ensure that this connectivity rule specifies that filter matches are not to be logged.

2. Otherwise, if you are manually configuring your IPSec policy:
   a. Ensure that the first IpFilterRule statement whose associated IpService statement has a Routing specification of Routed or Either permits all IPv4 addresses, permits all protocols and all security classes, and has a Direction specification of Bidirectional.
   
   **Tip:** If your rule has a Routing specification of Either, it applies to both local and routed traffic. If your security policy does not allow you to permit all local traffic, split this IpFilterRule into two filter rules, one with a Routing specification of Routed and one with a Routing specification of Local.
   b. Ensure that this IpFilterRule statement's associated IpGenericFilterAction statement does not specify an IpFilterLogging setting of Yes to enable filter logging.

For more information, see [QDIO Accelerator and IP security in z/OS Communications Server: IP Configuration Guide](http://www.ibm.com).

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP

Module
EZBISSFT, EZBISPSW, EZBIPEPR

Routing code
10
Automation
You can automate on this message to detect situations when routed traffic is not being QDIO accelerated because of IP filtering rules.

Example
EZD2021A QDIO ACCELERATOR IS ENABLED ONLY FOR SYSPLEX DISTRIBUTOR BECAUSE OF POLICY FILTER RULES

EZD2022A QDIO ACCELERATOR IS ENABLED ONLY FOR SYSPLEX DISTRIBUTOR BECAUSE OF DEFENSIVE FILTER RULES

Explanation
QDIO Accelerator will accelerate only Sysplex Distributor traffic. Routed traffic cannot be accelerated because the defensive filters managed by the defense manager daemon (DMD) are being used to block some routed traffic. To satisfy the defensive filters, routed traffic must be processed by the TCP/IP stack, where defensive filtering is implemented.

System action
Processing continues with QDIO Accelerator enabled only for sysplex distributor traffic.

Operator response
Contact the system programmer.

System programmer response
Defensive filters are usually added to block a temporary security condition such as an attack or scan. The defensive filters causing this condition are removed when their configured expiration time passes. You can wait for the filters to expire.

Before these defensive filters expire, you can work with the security administrator to evaluate whether the filters are still needed. If the defensive filters are no longer needed, the security administrator can use the `ipsec -F` command to delete them.

For more information, see QDIO Accelerator and IP security in z/OS Communications Server: IP Configuration Guide.

User response
Not applicable.

Problem determination
None.

Source
z/OS Communications Server TCP/IP

Module
EZBISDAD, EZBIPEPR
Routing code
10

Descriptor code
2

Automation
You can automate on this message to detect situations when routed traffic is not being QDIO accelerated because of defensive filter rules.

Example

<table>
<thead>
<tr>
<th>EZD2022A</th>
<th>QDIO ACCELERATOR IS ENABLED ONLY FOR SYSPLEX DISTRIBUTOR BECAUSE OF DEFENSIVE FILTER RULES</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD2023I</td>
<td>QDIO ACCELERATOR IS ENABLED WITH CURRENTLY INSTALLED IP FILTER RULES</td>
</tr>
</tbody>
</table>

Explanation
QDIO Accelerator is enabled for all traffic because the installed IP filters permit all routed traffic to pass through the associated TCP/IP stack. Therefore, that traffic can be accelerated without requiring special processing by the TCP/IP stack.

System action
Processing continues.

Operator response
No action is needed.

System programmer response
No action is needed.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBIPEPR, EZAPSCAN, EZBISDDL, EZBISPSW, EZBISSFT, EZBISEPR

Routing code
10

Descriptor code
12
Automation
You can automate on this message to detect situations when QDIO Acceleration is permitted by IP filtering rules.

Example

EZD2023I QDIO ACCELERATOR IS ENABLED WITH CURRENTLY INSTALLED IP FILTER RULES

EZD2024I type current maximum

Explanation
This message is issued as part of a message group. See message EZZ8453I in z/OS Communications Server: IP Messages Volume 4 (EZZ, SNM) for a complete description of the message group.

System action
See message EZZ8453I.

Operator response
See message EZZ8453I.

System programmer response
See message EZZ8453I.

User response
See message EZZ8453I.

Problem determination
See message EZZ8453I.

Source
See message EZZ8453I.

Module
See message EZZ8453I.

Routing code
See message EZZ8453I.

Descriptor code
See message EZZ8453I.

Automation
See message EZZ8453I.

Example
See message EZZ8453I.

EZD2025I ICSF services are currently unavailable to the IKE daemon
**Explanation**
The IKE daemon is initializing and ICSF is not active.
This message does not indicate an immediate problem. However, some IKE daemon requests can fail if ICSF remains inactive.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**User response**
None.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
ike_config.cpp

**Routing code**
2, 8

**Descriptor code**
12

**Automation**
This message is sent to the system console and to syslogd.

**Example**

```
EZD2025I ICSF services are currently unavailable to the IKE daemon
```

```
EZD2026I Tcpname GSK_FIPS_STATE_SET FAILED FOR AT-TLS GROUP group_name WITH RETURN CODE gsk_status description
```

**Explanation**
The AT-TLS group `group_name` is configured with one of the following FIPS140 values: On, Level1, Level2, Level3... `Tcpname` received an error from the System SSL Certificate Management Services (CMS) API gsk_fips_state_set. The gsk_fips_state_set API is documented in z/OS Cryptographic Services System SSL Programming. AT-TLS services are not available for `group_name`.

In the message text:
**Tcpname**

The name of the TCPIP stack

**group_name**

The name of the AT-TLS group that is specified on a TTLSGroupAction statement

**gsk_status**

The hexadecimal gsk_status code that is returned from gsk_fips_state_set()

**description**

Describes the meaning of gsk_status

**System action**

TCPIP continues. AT-TLS services are not available for the AT-TLS group specified by the `group_name` value.

**Operator response**

Contact the system programmer.

**System programmer response**

For more information about the error, see CMS status codes in z/OS Cryptographic Services System SSL Programming. If the CSFSERV resource class is active, review RACF CSFSERV resource requirements in z/OS Cryptographic Services System SSL Programming and ensure that the user ID for `Tcpname` has access to the appropriate resources in this class.

**User response**

None.

**Problem determination**

See the system programmer response.

**Source**

z/OS Communications Server TCP/IP

**Module**

EZBTLCMN

**Routing code**

2, 8

**Descriptor code**

12

**Automation**

This message goes to the console. Automation can notify the system programmer.

**Example**

```
EZD2026I TCPCS GSK_FIPS_STATE_SET FAILED FOR AT-TLS GROUP grp3 WITH RETURN CODE 0335308A Known Answer Test
has failed when attempting to use ICSF
```

1338  z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
**EZD2027I** Initiation of UDP encapsulated IKE version *major.minor* security association *generation* for tunnel *ID* is not permitted following DVIPA takeover; the remote peer is behind an NAPT, or is acting as a security gateway

**Explanation**
Negotiation of a UDP encapsulated security association (SA) following a dynamic virtual IP address (DVIPA) takeover was denied.

When performing UDP encapsulation, the z/OS host is limited to acting in responder mode when the remote peer is behind a network address port translation (NAPT) device, or is acting as a security gateway. See Configuration scenarios supported for NAT traversal in z/OS Communications Server: IP Configuration Guide for more information.

Additional diagnostic messages that have the same message instance number will be issued to identify the impacted SA. The message instance number precedes the message number in the log output and is used to group related messages from the Internet Key Exchange (IKE) daemon.

In the message text:

- **major.minor**
  The major and minor version of the IKE protocol for the SA.

- **generation**
  The number used to differentiate SAs for the same tunnel. The first SA created for a tunnel is number 1.

- **ID**
  The tunnel prefix and number used to identify the tunnel. The tunnel prefix is K for an IKE tunnel and Y for a dynamic tunnel.

**System action**
The SA for the DVIPA was not reestablished; IKE daemon processing continues.

**Operator response**
Examine the IKE syslog to determine the remote peer. Attempt to recover the SA by initiating the SA negotiation from the remote security endpoint. See Configuration scenarios supported for NAT traversal in z/OS Communications Server: IP Configuration Guide for more information.

**System programmer response**
None.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: IPSec

**Module**
CommonIPsecSA.cpp

**Routing code**
Not applicable for syslog message.
Descriptor code
Not applicable for syslog message.

Automation
This message goes to the syslog.

Example
EZD2027I Initiation of UDP encapsulated IKE version 2.0 security association 0 for tunnel Y0 is not permitted following DVIPA takeover; the remote peer is behind an NAPT or is acting as a security gateway

EZD2028I ResourceType ResourceName Activation failed – Reason

Explanation
A configuration error was detected during activation of the resource (device or interface).

In the message text:

ResourceType
The type of resource that failed activation. Possible values are device or interface.

ResourceName
The name of the device or interface.

Reason
Reason indicates the cause of the failure and can be one of the following:

An MPC group is defined as MPCUSAGE=EXC and is already in use
The MPC group that this user is trying to use is coded for exclusive use (MPCUSAGE=EXC), and it is already in use by another user.

IQDCHPID is not specified when multiple IQD CHPIDs are available
Multiple IQD (HiperSockets) CHPIDs are defined on this LPAR, and VTAM start option IQDCHPID does not specify which is to be used for dynamic XCF connectivity.

IQDIO devices are not available to build an MPC group
An attempt was made to build an IQDIO MPC group, but VTAM could not find at least three subchannel devices associated with the same IQD CHPID.

IQDIO IQD CHPID is in conflict with sysplex IQD CHPID
The user defined an IQDIO device CHPID on a TCP/IP DEVICE or INTERFACE statement and it conflicts with the sysplex IQD CHPID defined by the IQDCHPID VTAM start option for DYNAMICXCF communication.

Processor is not IQD capable
The processor does not support IQD CHPIDs.

IQDIO or OSA CHPID is not available to build an MPC group
An attempt to activate an IQDIO or OSA CHPID was rejected because the CHPID was not found.

- For IQDIO, one of the following conditions occurred:
  - When attempting dynamic XCF HiperSockets, either the VTAM start option IQDCHPID=None was coded, or the IQDCHPID value specified a CHPID that was not defined to the LPAR.
  - A TCP/IP HiperSockets DEVICE or INTERFACE statement specified a CHPID value that was not defined to the LPAR.
- For OSA-Express, the CHPID value specified on the TRLE TCP/IP DEVICE or INTERFACE statement was not defined to the LPAR.

OSA control channels are not available to build an MPC group
Two consecutive subchannel addresses, starting with an even number, are required when dynamically building a TRLE for OSA-Express devices that use queued direct I/O (QDIO). The consecutive subchannel...
addresses are required for the READ and WRITE control channels. The CHPID was found for the failing device, but two consecutive subchannel addresses starting with an even number were not found.

**Incorrect PORTNAME or TRLENAME is specified**
For devices and interfaces where the TRLE is defined in VTAM and not dynamically generated, the DEVICE or INTERFACE statement must point to a valid TRLE. Either the PORTNAME value, TRLENAME value, or DEVICE name does not match a corresponding value on a TRLE statement.

**PNETID is not configured**
An IBM 10 GbE RoCE Express feature configured on the GLOBALCONFIG statement does not have a physical network ID.

**No datapath device addresses are available**
A QDIO or iQDIO device or interface activation failed because there were no datapath devices available.

**The channel unit address is not available**
The device is attempting to use a channel unit address (CUA) that is not defined or not available.

**An incorrect channel unit address is specified**
The device is attempting to use an incorrect channel unit address (CUA).

**The channel unit address is already in use**
The channel unit address (CUA) is being used by other resources.

**QDIO CHPID type mismatch is detected**
The CHPID type for the OSA-Express device using QDIO does not match the type of CHPID that it is attempting to use.

**OSM/OSX activation not permitted because LPAR is not in Ensemble**
Activating OSM or OSX interface is not supported because the LPAR is not participating in an Ensemble.

**OSM/OSX activation not permitted because CPC is not in Ensemble**
Activating OSM or OSX devices is not supported because the CPC is not configured as a member of an Ensemble.

**IQD activation is not permitted against an IQDX device**
The DEVICE or INTERFACE statement for the failing resource specifies a HiperSockets CHPID that is defined with the Internal Queued Direct I/O extensions (IQDX) function of HiperSockets. A CHPID with the IQDX function is reserved for IQDX communications and cannot be specified on the DEVICE or INTERFACE statement.

**Portname is already in use by another port on this CHPID**
An attempt was made to activate an OSA-Express port in QDIO mode. The portname used on this activation attempt was already in use on the other port on this CHPID. Two ports on the same CHPID cannot have the same portname.

**Different portname is already assigned to OSA**
An attempt was made to activate an OSA-Express port in QDIO mode. The portname for this activation attempt did not match the portname already assigned to this port by a previous user. All z/OS users of a port must activate it with the same portname.

**IQD activation is not permitted against an IQDC device**
The DEVICE or INTERFACE statement for the failing resource specifies a HiperSockets CHPID that is defined with the Internal Queued Direct I/O extensions (External Bridge) function of HiperSockets. A CHPID with the External Bridge function is reserved for IQDC communications and cannot be specified on the DEVICE or INTERFACE statement.

**System action**
TCP/IP marks the device or interface inactive.

**Operator response**
Contact the system programmer.
System programmer response

Depending on the value of `Reason`, take the following actions:

**An MPC group is defined as MPCUSAGE=EXC and is already in use**
Ensure that there is only one user of a TRLE defined with MPCUSAGE=EXC. A user can be a TCP/IP device or interface, or a VTAM PU.

**IQDCHPID is not specified when multiple IQD CHPIIDs are available**
When more than one IQD CHPID is defined, VTAM start option IQDCHPID must specify the one that is to be used for dynamic XCF connectivity. Choose which CHPID is to be used for dynamic XCF, and specify it on the IQDCHPID VTAM start option. Also, ensure that the CHPID to be used for dynamic XCF is not used on a HiperSockets device of type MPCIPA, or a HiperSockets interface of type IPAQIDIO or IPAQIDIO6 for any stack on the LPAR. The CHPID specified on IQDCHPID is reserved for dynamic XCF connectivity.

**IQDIO devices are not available to build an MPC group**
Verify the Hardware Configuration Definition (HCD) or Input Output Control Data Set (IOCDS) configuration for accuracy for this logical partition. The HCD configuration might need to be updated to specify a sufficient number of HiperSockets devices.

**IQDIO IQD CHPID is in conflict with sysplex IQD CHPID**
- If dynamic XCF connectivity is required, ensure that no TCP/IP DEVICE or INTERFACE statement specifies the same CHPID that is defined for XCF connectivity on the IQDCHPID VTAM start option.
- If dynamic XCF connectivity is not required, ensure that DYNAMICXCF is not defined on the TCP/IP IPCONFIG or IPCONFIG6 statement, or specify IQDCHPID=NONE as a VTAM start option.

**Processor is not IQD capable**
Remove all TCP/IP HiperSockets DEVICE and INTERFACE statements. They are not available on this processor.

**IQDIO or OSA CHPID is not available to build an MPC group**
Ensure that the CHPID defined for the OSA-Express or HiperSockets DEVICE or INTERFACE is defined for this LPAR.

**OSA control channels are not available to build an MPC group**
Ensure that sufficient subchannel addresses exist for all users of the TRLE, and that at least two addresses are consecutive starting with an even number for the two control channels.

**Incorrect PORTNAME or TRLENAME is specified**
For the failing device or interface, determine which VTAM TRLE definition should be used.
- If the TRLE is identified on the INTERFACE statement by the PORTNAME parameter, ensure that the PORTNAME parameter contains a value that matches the PORTNAME parameter on the VTAM TRLE statement.
- If the TRLE is identified on the INTERFACE statement by the TRLENAME parameter, ensure that the TRLENAME parameter contains a value that matches the TRLE name of the VTAM TRLE statement.
- If the port name is identified on the DEVICE statement by a DEVICE name, ensure that the DEVICE name matches the PORTNAME parameter on the VTAM TRLE statement.

**PNETID is not configured**
Configure a physical network ID for the 10 GbE RoCE Express feature in HCD or remove the PCI-function ID (PFID) definition representing the 10 GbE RoCE Express feature from the SMCR parameter of the GLOBALCONFIG statement.

**No datapath device addresses are available**
Ensure that there are sufficient datapath device addresses defined for the device.
- When the TRLE is defined in VTAM, define sufficient DATAPATH addresses on the TRLE statement for all users of the TRLE. See DATAPATH in z/OS Communications Server: SNA Resource Definition Reference for more information about how many DATAPATH addresses are required for your configuration.
- When the TRLE is dynamically generated by VTAM, a certain number of DATAPATH addresses are dynamically allocated depending on the type of device and the number of addresses defined in HCD for the
device. See Resources automatically activated by VTAM in z/OS Communications Server: SNA Network Implementation Guide for more detail about how many DATAPATH addresses are allocated.

**The channel unit address is not available**
Verify that the channel unit address is defined in HCD or IOCDS and that it is varied online.

**An incorrect channel unit address is specified**
Verify that the channel unit address is correctly defined. Verify that both READ and WRITE CUAs are specified if they are both required.

**The channel unit address is already in use**
Verify that the device uses a channel unit address that is not in use by another user. Examples of other users are another TCP/IP stack on this LPAR, or VTAM.

**QDIO CHPID type mismatch is detected**
Verify that the CHPID type specified or defaulted on the configuration statement is the same as the type that is assigned to this device.

**OSM/OSX activation not permitted because LPAR is not in Ensemble**
Enable LPAR participation in the Ensemble by specifying the ENSEMBLE=YES VTAM start option.
See TCP/IP in an ensemble in z/OS Communications Server: IP Configuration Guide for more information about Ensembles.

**OSM/OSX activation not permitted because CPC is not in Ensemble**
If connectivity to the intraensemble data network or intranode management network is needed, ensure that the LPAR is configured as a member of the Ensemble. See TCP/IP in an ensemble in z/OS Communications Server: IP Configuration Guide for more information about Ensembles.

**IQD activation is not permitted against an IQDX device**
Either remove the HiperSockets DEVICE or INTERFACE statement, or specify a different HiperSockets CHPID. For more information about the IQDX function, see HiperSockets connectivity to the intraensemble data network in z/OS Communications Server: IP Configuration Guide.

**Portname is already in use by another port on this CHPID**
Identify the portname of the failing DEVICE or INTERFACE statement. It is specified as the DEVICE name on a DEVICE statement and as the PORTNAME value on an INTERFACE statement. Change the portname value on the DEVICE or INTERFACE statement and on the PORTNAME operand on the TRLE representing this port to a name that is unique on this CHPID.

**Different portname is already assigned to OSA**
Identify all users of the OSA-Express port and ensure that the portname is the same for all users. Users might be other TCP/IP stacks on this LPAR, or TCP/IP stacks on other LPARs.

**IQD activation is not permitted against an IQDC device**
Either remove the HiperSockets DEVICE or INTERFACE statement, or specify a HiperSockets CHPID that is not defined with the External Bridge function. For more information about the IQDC function, see HiperSockets Converged Interface overview in z/OS Communications Server: IP Configuration Guide.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: Configuration & Initialization

**Module**
EZBIFIUT
Routing code
2, 8

Descriptor code
12

Automation
Not applicable for automation.

Example

EZD2028I DEVICE IUTIQDIO ACTIVATION FAILED – IQDIO DEVICES ARE NOT AVAILABLE TO BUILD AN MPC GROUP
EZD2028I INTERFACE OSAQDIO46 ACTIVATION FAILED – INCORRECT PORTNAME or TRLENAME is SPECIFIED

EZD2029I   DCAS CONFIGURATION keyword value IS NOT SUPPORTED

Explanation
This message is issued when Digital Certificate Access Server (DCAS) is processing the DCAS configuration file, and the keyword value is not supported.

In the message text:

keyword
  The keyword that was specified

value
  The value that is not supported for the specified keyword

System action
DCAS ends.

Operator response
Contact the system programmer.

System programmer response
Correct the DCAS configuration file. See z/OS Communications Server: IP Configuration Reference for a list of the keywords and values that are supported for each keyword in the DCAS configuration file.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP

Module
dcasconf.c
Routing code
8, 10

Descriptor code
12

Automation
This message goes to the console. Automation can notify the system programmer to correct the DCAS configuration.

Example
EZD2029I DCAS CONFIGURATION TLSMECHANISM YES IS NOT SUPPORTED

EZD2030I TOO MANY VIPADEFINE AND VIPABACKUP VIPAS - ipaddr REJECTED

Explanation
There are already 1024 active and backup dynamic or moveable VIPAs on this stack. These VIPAs were defined using a combination of VIPADYNAMIC VIPADEFINE and VIPADYNAMIC VIPABACKUP statements on this stack and VIPADYNAMIC VIPADISTRIBUTE statement on other stacks. The specified IP address is not defined to the stack.

System action
TCP/IP continues.

Operator response
Correct the appropriate definitions and try the command or activation again.

System programmer response
Reduce the number of defined or backup Dynamic VIPAs for this stack or remove this stack as a distribution target used by other stacks.

User response
No action is needed.

Problem determination
See the System Programmer Response.

Source
z/OS Communications Server TCP/IP

Module
EZBXFDVI, EZBX6DVI

Routing code
2, 8
Automation

The message goes to the system console and the system log. This message indicates an error in the profile, which requires analysis and a proper response cannot be easily automated.

Example

<table>
<thead>
<tr>
<th>EZZ8320I TOO MANY VIPADEFINE AND VIPABACKUP VIPAS - 9.67.242.3 REJECTED</th>
</tr>
</thead>
</table>

**EZD2031I**  SMC APPLICABILITY TOOL HAS STARTED COLLECTING DATA

Explanation

This message is a result of the VARY TCPIP,,SMCAT command and indicates that the SMC applicability tool has started collecting data.

System action

The system begins to collect data for the SMCAT report and displays the VARY TCPIP,,SMCAT command configuration parameters by issuing message EZD2040I to the system log and job log.

Operator response

Do one of the following actions:

- Wait until the current data collection interval expires. To determine when the interval expires, review the console messages to determine when the tool was turned on.
- Use the VARY TCPIP,,SMCAT,OFF command to turn the tool off.

See "VARY TCPIP,,SMCAT" in z/OS Communications Server: IP System Administrator's Commands for more information about the SMC applicability tool.

System programmer response

No action is needed.

User response

No action is needed.

Problem determination

Not applicable.

Source

z/OS Communications Server TCP/IP

Module

EZBTCIC2

Routing code

2, 8
**Descriptor code**
12

**Automation**
Not applicable for automation.

**Example**

```
EZD2031I SMC APPLICABILITY TOOL HAS STARTED COLLECTING DATA
```

**EZD2032I SMC APPLICABILITY TOOL HAS STOPPED COLLECTING DATA**

**Explanation**
This message is a result of the `VARY TCPIP,,SMCAT,OFF` command or is a result of the expiration of the time interval specified on a previous `VARY TCPIP,,SMCAT` command. It indicates that the SMC applicability tool has stopped collecting data.

**System action**
The system generates the SMCAT report and issues message EZD2033I to the system log and job log.

**Operator response**
Inform the system programmer that the SMCAT report is available to be analyzed.

**System programmer response**
Review the data in message EZD2033I. See the Report Examples portion of the “VARY TCPIP,,SMCAT” section in z/OS Communications Server: IP System Administrator’s Commands for assistance in understanding the output.

**User response**
No action is needed.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP

**Module**
EZBTCIC2

**Routing code**
2, 8

**Descriptor code**
12

**Automation**
Not applicable for automation.
Example

EZD2032I SMC APPLICABILITY TOOL HAS STOPPED COLLECTING DATA

EZD2033I TCP/IP CS versionRelease TCPIP Name: name

Explanation
This is the first message in the VARY TCPIP,,SMCAT command report. This report is written to the system log and the job log. See the information about the "VARY TCPIP,,SMCAT" command in z/OS Communications Server: IP System Administrator's Commands for a detailed description of the report.

In the message text:

versionRelease
The z/OS Communications Server version and release.

name
The name of the TCP/IP stack.

System action
Not applicable.

Operator response
No action is needed.

System programmer response
Analyze the report to determine the benefits that SMC-R can provide for your workload. See the Report Examples section of "VARY TCPIP,,SMCAT" in z/OS Communications Server: IP System Administrator's Commands for assistance in understanding the output.

User response
No action is needed.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBTCIC2

Routing code
11

Descriptor code
6

Automation
Not applicable for automation.
EZD2034I   name server IS NOT SUPPORTED

Explanation
The z/OS Communications Server server indicated by name, is not supported.
In the message text:
name
   The name of the server that is not supported.

System action
The server ends.

Operator response
See the z/OS Communications Server documentation for information about the server.

System programmer response
See the z/OS Communications Server documentation for information about the server.

User response
See the z/OS Communications Server documentation for information about the server.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
Not applicable.

Routing code
10

Descriptor code
*

Automation
Not applicable for automation.

Example
EZD2034I SNALINK_LU0 SERVER IS NOT SUPPORTED

EZD2035I   NAME SERVER ipaddress
**Message Format:**

NAME SERVER *ipaddress*

**STATUS:** *status*  
**FAILURE RATE:** *percent*

**Explanation**

This is a multi-line message that is issued by the resolver in response to a MODIFY RESOLVER,DISPLAY command or a MODIFY RESOLVER,REFRESH command when the autonomic quiescing of unresponsive name servers function is active. The resolver issues a message for each name server specified on an NSINTERADDR statement in the global TCPIP.DATA file. The message displays the status and the DNS query failure rate for the specified name server.

In the message text:

**ipaddress**

The IPv4 or IPv6 network address of the name server.

**status**

The status of the name server specified by the *ipaddress* value. Possible values are:

**ACTIVE**

The resolver considers the name server to be responsive to the DNS queries that are generated by an application or by the resolver. The resolver continues to forward the DNS queries that are generated by an application to this name server.

**QUIESCED**

The resolver considers the name server to be unresponsive to the DNS queries that are generated by an application or by the resolver. Unless all name servers are quiesced, the resolver does not forward the DNS queries that are generated by an application to this name server. If all name servers are quiesced, the resolver sends the DNS queries that are generated by an application to the quiesced name servers, instead of immediately failing the query.

**percent**

The DNS query failure rate that was calculated by the resolver at the last monitoring checkpoint for the name server specified by the *ipaddress* value. The DNS query failure rate is the sum of the DNS queries that are generated by applications or by the resolver that received no response divided by the total number of DNS queries that are generated by an application or by the resolver that were sent during a checkpoint interval. The percent value is a value in the range 0% - 100%. The value *N/A* is displayed when the resolver was unable to calculate a DNS query failure rate at the last monitoring checkpoint. The resolver might be unable to calculate a DNS query failure rate for the following reasons:

- The resolver sent less than total ten DNS queries to the name server. The resolver requires a sample set of ten DNS queries during a monitoring interval to declare that a name server is unresponsive.
- The resolver has detected recent system changes, such as activation of the TCPIP stack, which might affect name server responsiveness. The resolver delays declaring that a name server is unresponsive for a predefined period of time after such system changes.

**System action**

Processing continues.

**Operator response**

- Contact the system programmer if one or more name servers display the QUIESCED status.
- Contact the system programmer if the network is experiencing delays and one or more name servers display the ACTIVE status with a nonzero DNS query failure rate.

**System programmer response**

If the status of a name server is QUIESCED, perform one of the following actions:
• If a network condition is preventing resolver requests or name server responses from reaching the correct
destination, correct the network condition. When the name server successfully responds to resolver polling
queries, the resolver will resume sending DNS queries that are generated by an application to the name server.
• If a configuration error is causing the name server to be unresponsive, use resolver diagnostic tools such as
the MODIFY RESOLVER,DISPLAY command or the Trace Resolver output to determine which of the following
conditions is causing the error:
  – If the IP address is no longer valid as a name server, remove the IP address from the list of name servers to
    be used by the resolver. The list of name servers is defined by using the NSINTERADDR or NAMESERVER
    configuration statement in the global TCPIP.DATA file.
  – If the RESOLVERTIMEOUT value is too low for responses to consistently return from the name server within
    the specified time value, increase the timeout setting to a value that permits a larger percentage of
    responses to arrive within the timeout interval.
After you correct the configuration error, instruct the operator to issue the MODIFY RESOLVER,REFRESH
command.
• If you eliminate a network condition and a configuration error as the reason for the message, the resolver
might be generating the message for a temporary condition that might resolve itself. For example, the name
server might be having maintenance applied, or the name server might have a very high percentage of failures
because few queries were sent to the name server during the monitoring interval. Even a short network
interruption can severely impact the calculations. If this situation repeats itself, an overly aggressive
UNRESPONSIVETHRESHOLD value might be contributing to the situation. Consider increasing the setting value
for the UNRESPONSIVETHRESHOLD parameter in the resolver setup file, and then instruct the operator to
issue the MODIFY RESOLVER,REFRESH,SETUP=setup_file_name command to make the resolver less sensitive
to name server response failures.
• If you want the resolver to resume forwarding the DNS queries that are generated by an application to the
name server, even if the name server is unresponsive at the current UNRESPONSIVETHRESHOLD percentage,
perform one of the following actions:
  – Increase the UNRESPONSIVETHRESHOLD percentage in the resolver setup file to be greater than the
    displayed DNS query failure rate. For example, if the UNRESPONSIVETHRESHOLD percentage is 10%, but
    the name server is failing to respond to 15% of the queries, increase the threshold to a value in the range
    20%. The resolver will continue to monitor the responsiveness of name servers in your network, but will
    forward DNS queries to this name server as long as the failure rate stays below the new threshold
    percentage.
  – Delete the AUTOQUIESCE operand from the UNRESPONSIVETHRESHOLD statement in the resolver setup
    file. The resolver will continue to monitor the responsiveness of name servers in your network, but will alert
    only the network operator of the unresponsive condition, and will continue to send DNS queries to all name
    servers.
After you modify the resolver setup file, instruct the operator to issue the MODIFY
RESOLVER,REFRESH,SETUP=setup_file_name command.
If the status of a name server is ACTIVE with a nonzero DNS query failure rate, your network is experiencing
network delays. The diagnosis of the network failures indicates that the DNS query failures are causing the
network delays. You might want to decrease the unresponsive threshold percentage. A lower percentage setting
causes the resolver to detect unresponsive name servers more rapidly and to stop forwarding the DNS queries
that are generated by an application to those name servers sooner. Decrease the threshold percentage by coding
a lower value on the UNRESPONSIVETHRESHOLD statement in the resolver setup file. You must specify a value
lower than the DNS query failure rate displayed for the name server. After you modify the
UNRESPONSIVETHRESHOLD statement, instruct the operator to issue the MODIFY
RESOLVER,REFRESH,SETUP=setup_file_name command.

User response
Not applicable.
Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: System Resolver

Module
ezbrecfg

Routing code
2, 8

Descriptor code
12

Automation
This message is not a good candidate for automation.

Example
F RESOLVER,DISPLAY
EZ2036I  AUTOQUIESCE IGNORED - GLOBALTCPIPDATA REQUIRED

Explanation
The resolver issues this message when the AUTOQUIESCE operand is specified on the UNRESPONSIVETHRESHOLD resolver setup statement, but no GLOBALTCPIPDATA resolver setup statement is coded. The AUTOQUIESCE operand setting is ignored.

System action
The resolver uses the UNRESPONSIVETHRESHOLD percentage value to perform the network operator notification function instead of performing the autonomic quiescing of unresponsive name servers function.

Operator response
Contact the system programmer.

System programmer response
If you do not want unresponsive name servers to be automatically quiesced, perform one of the following actions:
• Remove the AUTOQUIESCE operand from the UNRESPONSIVETHRESHOLD statement, but leave the threshold percentage coded on the statement.

• Remove the UNRESPONSIVETHRESHOLD statement completely. The network operator notification function will run by default.

• Leave the resolver setup file unchanged. You will continue to see message EZD2036I every time the resolver is started or a MODIFY RESOLVER,REFRESH command is issued, but the autonomic quiescing function will not be active.

If you want unresponsive name servers to be automatically quiesced, perform the following actions:

• If you do not have a global TCPIP.DATA file, create one. Code the appropriate resolver-related TCPIP.DATA statements in the global TCPIP.DATA file you just created.

• Code the GLOBALTCPIPDATA statement in the resolver setup file, specifying the name of the global TCPIP.DATA file to be used.

If you have corrected the resolver setup file, instruct the operator to issue the MODIFY RESOLVER,REFRESH,SETUP=setup_file_name command to activate the changes.

User response
None.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: System Resolver

Module
EZBRECFG

Routing code
2, 8

Descriptor code
12

Automation
This message is not a good candidate for automation.

Example

<table>
<thead>
<tr>
<th>EZD2036I</th>
<th>AUTOQUIESCE IGNORED - GLOBALTCPIPDATA REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZD2037E</td>
<td>resolver_setup_value PROCESSING ENDED - ISSUE RESOLVER REFRESH TO REACTIVATE</td>
</tr>
</tbody>
</table>

Explanation

This message is issued when the resolver function defined by the resolver_setup_value value ends abnormally. The operator can restart the resolver function by issuing a MODIFY RESOLVER,REFRESH or MODIFY RESOLVER,REFRESH,SETUP= command.

Possible values of resolver_setup_value are:

• AUTOQUIESCE
System action

Processing continues.

- If the \texttt{resolver\_setup\_value} value is AUTOQUIESCE, the resolver stops performing the autonomic quiescing of unresponsive name server function, and instead performs the network operator notification of unresponsive name server function. All name servers previously identified by message EZZ9311E in \textit{z/OS Communications Server: IP Messages Volume 4 (EZZ, SNM)} as being unresponsive are now considered responsive, and the resolver will now send the DNS queries that are generated by an application to these name servers. If the resolver determines subsequently that the name server is still unresponsive, the resolver will generate message EZZ9308E in \textit{z/OS Communications Server: IP Messages Volume 4 (EZZ, SNM)} to notify the network operator.

- If the \texttt{resolver\_setup\_value} value is UNRESPONSIVETHRESHOLD, the resolver stops performing all monitoring of unresponsive name servers. All name servers previously identified by message EZZ9308E as being unresponsive are now considered responsive. All name servers previously identified by message EZZ9311E as being unresponsive are now considered responsive, and the resolver will now send DNS queries that are generated by an application to these name servers. The resolver will not generate any additional messages EZZ9308E or EZZ9311E.

The message remains on the operator console until one of the following events occurs:

- The operator issues a MODIFY RESOLVER,REFRESH command. The resolver restarts the function at the previously defined level of functionality.
- The operator issues a MODIFY RESOLVER,REFRESH,SETUP=\texttt{resolver\_setup\_file} command. The resolver activates the level of function defined in the \texttt{resolver\_setup\_file} file.
- The resolver is stopped.

Operator response

If you want the resolver to use the same level of monitoring function, issue a MODIFY RESOLVER,REFRESH command. If not, contact the system programmer.

System programmer response

If you want the resolver to use the same level of function, instruct the operator to issue a MODIFY RESOLVER,REFRESH command.

If you want the resolver to use a different level of function, first modify your resolver setup file.

- If you want the resolver to automatically quiesce unresponsive name servers, specify the AUTOQUIESCE operand on the UNRESPONSIVETHRESHOLD resolver setup statement. When the AUTOQUIESCE operand is specified, the resolver automatically stops sending the DNS queries generated by an application to unresponsive name servers. The resolver uses the threshold value specified on the UNRESPONSIVETHRESHOLD resolver setup statement to determine whether a name server is unresponsive.
- If you no longer want the resolver to automatically quiesce unresponsive name servers, delete the AUTOQUIESCE operand from the UNRESPONSIVETHRESHOLD resolver setup statement. The resolver continues to monitor the responsiveness of name servers in your network, but only alerts the network operator of the unresponsive condition. The resolver continues to send the DNS queries generated by an application to all name servers.
- If you no longer want the resolver to perform any level of monitoring of unresponsive name servers, specify the UNRESPONSIVETHRESHOLD(0) statement in the resolver setup file.

After you modify the resolver setup file, instruct the operator to issue the MODIFY RESOLVER,REFRESH,SETUP=\texttt{setup\_file\_name} command.

User response

Not applicable.
Problem determination

In most cases, the resolver will generate a dump in addition to displaying this message. Contact IBM Service with the dump information for additional problem determination.

Source

z/OS Communications Server TCP/IP: System Resolver

Module

EZBREUPS, EZBRENSR

Routing code

2, 8

Descriptor code

2

Automation

This message is a good candidate for automation, if you want to restart the same level of resolver function by using the MODIFY RESOLVER,REFRESH command.

Example

EZD2037E AUTOQUIESCE PROCESSING ENDED - ISSUE RESOLVER REFRESH TO REACTIVATE

EZD2038I RESOLVER INITIALIZATION COMPLETED WITH WARNINGS

Explanation

The resolver detected errors in setup statements in the resolver setup file and issued warning messages during the initialization of the resolver address space. The resolver address space has initialized and is ready to accept MODIFY and STOP commands, and resolver services are available to applications.

System action

Processing continues.

Operator response

Save the system log and contact the system programmer.

System programmer response

Examine the system log to find the warning messages that are issued during the initialization of the resolver address space.

- If the errors in the resolver setup file did not change the resolver configuration settings from what you wanted, you can ignore the errors. If, at a later time, you want to use the same setup file as part of the MODIFY RESOLVER,REFRESH,SETUP command processing, correct the errors at that time.

- If the errors in the resolver setup file changed the resolver configuration settings from what you wanted, correct the errors in the setup file. See the system programmer actions defined for the warning messages that are issued by the resolver to determine the corrective actions to take. After you have corrected the errors, instruct the operator to issue a MODIFY RESOLVER,REFRESH,SETUP command to correct the resolver configuration settings.
**User response**
No action is needed.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: System Resolver

**Module**
EZBREINI

**Routing code**
2, 8

**Descriptor code**
5

**Automation**
This message is displayed at the operator console and is suitable for automation. You can use it to detect resolver setup file errors that might impact your system operations.

**Example**

```
EZD2038I  RESOLVER INITIALIZATION COMPLETED WITH WARNINGS
```

**EZD2039I  WARNINGS ISSUED DURING RESOLVER INITIALIZATION**

**Explanation**
The resolver includes this message in the response to a `MODIFY RESOLVER,DISPLAY` command if the resolver issued warning messages during the initialization of the resolver address space and you did not subsequently issue a `MODIFY RESOLVER,REFRESH,SETUP` command to correct the errors.

**System action**
Processing continues.

**Operator response**
Save the resolver configuration settings and contact the system programmer.

**System programmer response**
Examine the resolver configuration settings.

- If the resolver configuration settings are accurate, you can ignore this message.
- If the resolver configuration settings are not accurate, examine the system log from the time when the resolver address space initialized to determine which warning messages the resolver issued. See the system programmer response for those warning messages to determine the corrective actions to take. After you correct the errors in the setup file, instruct the operator to issue a `MODIFY RESOLVER,REFRESH,SETUP` command to correct the resolver configuration settings.
User response
No action is needed.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: System Resolver

Module
EZBREINI

Routing code
2, 8

Descriptor code
5

Automation
This message is displayed at the operator console and is suitable for automation. You can use it to detect resolver setup file errors that might impact your system operations.

Example

F RESOLVER,DISPLAY
EZ9298I DEFAULTTCPIPDATA - None
EZ9298I GLOBALTCPIPDATA - None
EZ9298I DEFAULTIPNODES - None
EZ9298I GLOBALIPNODES - None
EZ9304I NOCOMMONSEARCH
EZ9304I CACHE
EZ9298I CACHESIZE = 200M
EZ9298I MAXTTL = 2147483647
EZ9298I UNRESPONSIVETHRESHOLD = 25
EZ2039I WARNINGS ISSUED DURING RESOLVER INITIALIZATION
EZ9293I DISPLAY COMMAND PROCESSED

EZD2040I  TCP/IP CS versionRelease TCPIP Name: name

Explanation
This is the first message in the display of configuration parameters from a successful VARY TCPIP,,SMCAT command. This message is written to the system log and the job log. See the information about the "VARY TCPIP,,SMCAT" command in z/OS Communications Server: IP System Administrator's Commands for a detailed description of the configuration parameters.

In the message text:

versionRelease
   The z/OS Communications Server version and release.

name
   The name of the TCP/IP stack.

System action
The system reports the configuration parameters which will be used for the VARY TCPIP,,SMCAT command report.
Operator response
No action is needed.

System programmer response
No action is needed.

User response
No action is needed.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBTCIC2

Routing code
11

Descriptor code
6

Automation
Not applicable for automation.

Example
EZD2040I TCP/IP CS V2R2 TCPIP Name: TCPCS1
Interval: 60 minutes
IP addresses/subnets being monitored
9.1.1.1/24
9.1.2.1/24
2001::1/116

EZD2041I IP ADDRESS ipaddr ALREADY SPECIFIED

Explanation
A duplicate IP address was detected in the data set specified on the VARY TCPIP,,SMCAT command.

In the message text:

ipaddr
The IP address.

System action
The system stops processing the VARY TCPIP,,SMCAT command.
Operator response
Correct the configuration error in the data set and reissue the **VARY TCPIP,,SMCAT** command.

System programmer response
No action is needed.

User response
No action is needed.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP

Module
EZBTCIC2

Routing code
2, 8

Descriptor code
12

Automation
Not applicable for automation.

Example

```
EZD2041I IP ADDRESS 201.0.1.1 ALREADY SPECIFIED
```

**EZD2042I**

*Tcpname FAILURE DETECTED IN AT-TLS GROUP group_name GRPID gid RSN rsncode*

Explanation
Application Transparent Transport Layer Security (AT-TLS) detected a failure in the AT-TLS group that is identified by the group name and group ID.

In the message text:

*Tcpname*
The name of the TCP/IP stack where the failure was detected

*group_name*
The name of the AT-TLS group that is specified on a TTLSGroupAction statement

*gid*
A hexadecimal value that uniquely identifies the AT-TLS group that supports the connection or SSL environment

*rsncode*
An IBM internal reason code
System action
TCP/IP continues. The AT-TLS group group_name is marked as failed. Any connections that attempt to use that group are reset. AT-TLS sets a return code of 5018 for those connections.

Operator response
Save the system log for problem determination and contact the system programmer. A Policy Agent AT-TLS policy refresh with the FLUSH option specified might recover the failed AT-TLS group for use by future connections. For more information about the effect of FLUSH option for refreshing AT-TLS policies, see FLUSH and PURGE considerations in z/OS Communications Server: IP Configuration Guide.

System programmer response
Collect a console dump that contains the TCP/IP address space and contact IBM Software Support.

User response
None.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP

Module
EZBTLMMSG

Routing code
10

Descriptor code
12

Automation
This message goes to the console. Automation can notify the system programmer.

Example
EZD2042I TCPCS FAILURE DETECTED IN AT-TLS GROUP gact1 GRPID 00000001 RSN 000003E9


Explanation
Application Transparent Transport Layer Security (AT-TLS) detected a failure during the invocation of the CELAAUTH macro for a pre-initialized environment while AT-TLS processes the specified event.

In the message text:

*gid*
- The hexadecimal value that identifies the AT-TLS group that supports the connection or SSL environment.
The hexadecimal value that identifies the AT-TLS environment that supports the connection. Multiple AT-TLS environments might be represented by a single master System SSL secure environment. If $eid$ is 00000000, the event does not apply to a specific environment.

cid
A hexadecimal value that identifies this TCP connection for the life of the connection. If $cid$ is 00000000, the event does not apply to a specific TCP connection.

loc_ip
The local IPv4 or IPv6 address.

loc_port
The local port number.

rem_ip
The remote IPv4 or IPv6 address.

rem_port
The remote port number.

jobname
The job name of the application that is associated with this TCP connection.

uesrid
The user ID of the application that is associated with this TCP connection.

rule
The name of the AT-TLS rule that mapped this TCP connection.

rcode
The CELAAUTH macro return code that indicates why the event failed. The $rcode$ values are defined in z/OS Language Environment®.

rsncode
The CELAAUTH macro reason code that indicates why the event failed. The $rsncode$ values are defined in z/OS Language Environment.

event
The AT-TLS event that was in process when the error occurred. Possible values are:

CELA READ
The TCP connection was attempting to decrypt secure data.

CELA WRITE
The TCP connection was attempting to encrypt secure data.

CELA SHUTDOWN
The TCP connection was attempting to shutdown the secure session.

CELA VALIDATEHOST
The TCP connection was attempting to validate the host name that was specified in a peer X.509 certificate.

System action:
TCP/IP continues. The pre-initialized environment that encountered the failure is stopped. The TCP connection that is associated with the specified event is ended.

Operator response
None.

System programmer response
The message creation time and owning TCP/IP job name of the process that created this message are included in the syslog trace before the message ID. The message has a syslog priority of ERROR and is written to the syslog when AT-TLS trace option ERROR(2) is specified.
Collect the syslogd log and a console dump that contains the TCP/IP address space and contact IBM Software Support.

**User response:**
None.

**Problem determination:**
See the system programmer response.

**Source:**
z/OS Communications Server TCP/IP

**Module**
EZBTLMSG

**Routing code**
2

**Descriptor code**
8

**Automation**
This message is not a candidate for automation.

**Example**

```
```

**EZD2044I** A refresh of IKEv2 security association *sa_generation* for tunnel *tunnel_id* was not scheduled due to the expiration of the remote security endpoint certificate in *secs* seconds

**Explanation**
This message is issued when a refresh for an Internet Key Exchange version 2 (IKEv2) Security Association (SA) was not scheduled because the lifetime of the SA expires at the same time the certificate used to authenticate the remote security endpoint expires. This message indicates that the remote security endpoint needs to re-authenticate by creating a new IKE SA and using a different certificate. If the remote security endpoint attempts to reuse the same certificate, message EZD1038I will be issued.

In the message text:

- **sa_generation**
  The number used to differentiate SAs for the same tunnel. The first SA that is created for a tunnel is number 1.
- **tunnel_id**
  The tunnel prefix and number used to identify the tunnel. The tunnel prefix is K for an IKE tunnel and Y for a dynamic tunnel.
- **secs**
  The number of seconds remaining before the remote security endpoint certificate expires.

**System action:**
IKE daemon processing continues.

**Operator response**
None.
**System programmer response:**
None.

**User response:**
Not applicable.

**Problem determination:**
None.

**Source:**
z/OS Communications Server TCP/IP: IKE daemon

**Module**
CommonIKESA.cpp

**Routing code**
11

**Descriptor code**
7

**Automation**
This message is output to syslog.

**Example**

```
EZD2044I A refresh of IKEv2 security association 2 for tunnel K9 was not scheduled due to the expiration of the remote security endpoint certificate in 1200 seconds
```

**EZD2046I** A create signature request to the NSS server failed; a matching certificate for local security endpoint identity `local_id` and authentication method `auth_method` was found but was not signed by a CA requested by the peer.

**Explanation**

The IKE daemon made a network security certificate services request to create a digital signature. A certificate matching the local security endpoint identity and authentication method was found on the network security services (NSS) server key ring, but the certificate was not signed by a certificate authority (CA) requested by the peer.

The IKE peer can send one or more certificate request payloads which requests that the local IKE send a certificate signed by one of the specified CAs. For an IKEv1 negotiation, if a certificate is found that matches the local security endpoint identity but is not signed by a specified CA, the create signature request fails. Message EZD2046I is generated to identify this failure.

In the message text:

- `local_id` - The local security endpoint identity
- `auth_method` - The authentication method

**System action**
The request fails; the IKE daemon continues.
Operator response
Contact the system programmer.

System programmer response
Determine the CA that signed the matching certificate on your local key ring. Contact the administrator of the remote security endpoint to determine what CAs are being requested. Either the peer must be updated to accept a certificate signed by the CA that signed the local certificate or a certificate must be added to the local NSS key ring that is signed by a CA acceptable to the peer.

User response
Not applicable.

Problem determination
You can perform the following steps to determine the CAs being requested by the remote peer:

1. Update IkeSyslogLevel in the IKED configuration file to include level 8, IKE_SYSLOG_LEVEL_FMTPKTTRC, in addition to any existing levels that you have enabled.
2. Retry the security association negotiation.
3. Review the IKED syslogd output. If the peer is sending one or more certificate request payloads, each will be labeled "Certificate Request Payload" in a formatted receive message in the syslogd output. The certificate request payloads identify the CAs that the peer is requesting.

Source:
z/OS Communications Server TCP/IP: IKE daemon

Module
CommonIKESA.cpp

Routing code
10

Descriptor code
12

Automation
Not applicable.

Example
EZD2046I A create signature request to the NSS server failed; a matching certificate for local security endpoint identity 1.2.3.4 and authentication method RsaSignature was found but was not signed by a CA requested by the peer.
This appendix lists the related protocol specifications (RFCs) for TCP/IP. The Internet Protocol suite is still evolving through requests for comments (RFC). New protocols are being designed and implemented by researchers and are brought to the attention of the Internet community in the form of RFCs. Some of these protocols are so useful that they become recommended protocols. That is, all future implementations for TCP/IP are recommended to implement these particular functions or protocols. These become the de facto standards, on which the TCP/IP protocol suite is built.


Draft RFCs that have been implemented in this and previous Communications Server releases are listed at the end of this topic.

Many features of TCP/IP Services are based on the following RFCs:

RFC
Title and Author
---
RFC 652
Telnet output carriage-return disposition option D. Crocker

RFC 653
Telnet output horizontal tabstops option D. Crocker

RFC 654
Telnet output horizontal tab disposition option D. Crocker

RFC 655
Telnet output formfeed disposition option D. Crocker

RFC 657
Telnet output vertical tab disposition option D. Crocker

RFC 658
Telnet output linefeed disposition D. Crocker

RFC 698
Telnet extended ASCII option T. Mock

RFC 726
Remote Controlled Transmission and Echoing Telnet option J. Postel, D. Crocker

RFC 727
Telnet logout option M.R. Crispin

RFC 732
Telnet Data Entry Terminal option J.D. Day

RFC 733
Standard for the format of ARPA network text messages D. Crocker, J. Vittal, K.T. Pogran, D.A. Henderson

RFC 734
SUPDUP Protocol M.R. Crispin

RFC 735
Revised Telnet byte macro option D. Crocker, R.H. Gumpertz

RFC 736
Telnet SUPDUP option M.R. Crispin

RFC 749
Telnet SUPDUP—Output option B. Greenberg

RFC 765
File Transfer Protocol specification J. Postel
RFC 885  
Telnet end of record option J. Postel

RFC 894  
Standard for the transmission of IP datagrams over Ethernet networks C. Hornig

RFC 896  
Congestion control in IP/TCP internetworks J. Nagle

RFC 903  
Reverse Address Resolution Protocol R. Finlayson, T. Mann, J. Mogul, M. Theimer

RFC 904  
Exterior Gateway Protocol formal specification D. Mills

RFC 919  
Broadcasting Internet Datagrams J. Mogul

RFC 922  
Broadcasting Internet datagrams in the presence of subnets J. Mogul

RFC 927  
TACACS user identification Telnet option B.A. Anderson

RFC 933  
Output marking Telnet option S. Silverman

RFC 946  
Telnet terminal location number option R. Nedved

RFC 950  
Internet Standard Subnetting Procedure J. Mogul, J. Postel

RFC 952  
DoD Internet host table specification K. Harrenstien, M. Stahl, E. Feinler

RFC 959  
File Transfer Protocol J. Postel, J.K. Reynolds

RFC 961  
Official ARPA-Internet protocols J.K. Reynolds, J. Postel

RFC 974  
Mail routing and the domain system C. Partridge

RFC 1001  

RFC 1002  

RFC 1006  
ISO transport services on top of the TCP: Version 3 M.T. Rose, D.E. Cass

RFC 1009  
Requirements for Internet gateways R. Braden, J. Postel

RFC 1011  
Official Internet protocols J. Reynolds, J. Postel

RFC 1013  
X Window System Protocol, version 11: Alpha update April 1987 R. Scheifler

RFC 1014  
XDR: External Data Representation standard Sun Microsystems

RFC 1027  
Using ARP to implement transparent subnet gateways S. Carl-Mitchell, J. Quarterman
<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
<th>Author(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1112</td>
<td>Host extensions for IP multicasting</td>
<td>S.E. Deering</td>
</tr>
<tr>
<td>1113</td>
<td>Privacy enhancement for Internet electronic mail: Part I — message</td>
<td>J. Linn</td>
</tr>
<tr>
<td></td>
<td>encipherment and authentication procedures</td>
<td></td>
</tr>
<tr>
<td>1118</td>
<td>Hitchhikers Guide to the Internet</td>
<td>E. Krol</td>
</tr>
<tr>
<td>1122</td>
<td>Requirements for Internet Hosts—Communication Layers</td>
<td>R. Braden, Ed.</td>
</tr>
<tr>
<td>1123</td>
<td>Requirements for Internet Hosts—Application and Support</td>
<td>R. Braden, Ed.</td>
</tr>
<tr>
<td>1146</td>
<td>TCP alternate checksum options</td>
<td>J. Zweig, C. Partridge</td>
</tr>
<tr>
<td>1155</td>
<td>Structure and identification of management information for TCP/IP-based internets</td>
<td>M. Rose, K. McCloghrie</td>
</tr>
<tr>
<td>1156</td>
<td>Management Information Base for network management of TCP/IP-based internets</td>
<td>K. McCloghrie, M. Rose</td>
</tr>
<tr>
<td>1158</td>
<td>Management Information Base for network management of TCP/IP-based internets: MIB-II</td>
<td>M. Rose</td>
</tr>
<tr>
<td>1166</td>
<td>Internet numbers</td>
<td>S. Kirkpatrick, M.K. Stahl, M. Recker</td>
</tr>
<tr>
<td>1179</td>
<td>Line printer daemon protocol</td>
<td>L. McLaughlin</td>
</tr>
<tr>
<td>1180</td>
<td>TCP/IP tutorial</td>
<td>T. Socolofsky, C. Kale</td>
</tr>
<tr>
<td>1183</td>
<td>New DNS RR Definitions</td>
<td>C.F. Everhart, L.A. Mamakos, R. Ullmann, P.V. Mockapetris</td>
</tr>
<tr>
<td>1184</td>
<td>Telnet Linemode Option</td>
<td>D. Borman</td>
</tr>
<tr>
<td>1186</td>
<td>MD4 Message Digest Algorithm</td>
<td>R.L. Rivest</td>
</tr>
<tr>
<td>1187</td>
<td>Bulk Table Retrieval with the SNMP</td>
<td>M. Rose, K. McCloghrie, J. Davin</td>
</tr>
<tr>
<td>1188</td>
<td>Proposed Standard for the Transmission of IP Datagrams over FDDI Networks</td>
<td>D. Katz</td>
</tr>
<tr>
<td>1190</td>
<td>Experimental Internet Stream Protocol: Version 2 (ST-II)</td>
<td>C. Topolcic</td>
</tr>
<tr>
<td>1191</td>
<td>Path MTU discovery</td>
<td>J. Mogul, S. Deering</td>
</tr>
<tr>
<td>1198</td>
<td>FYI on the X window system</td>
<td>R. Scheifler</td>
</tr>
<tr>
<td>1207</td>
<td>FYI on Questions and Answers: Answers to commonly asked “experienced Internet user” questions</td>
<td>G. Malkin, A. Marine, J. Reynolds</td>
</tr>
<tr>
<td>1208</td>
<td>Glossary of networking terms</td>
<td>O. Jacobsen, D. Lynch</td>
</tr>
</tbody>
</table>
RFC 1213
Management Information Base for Network Management of TCP/IP-based internets: MIB-II K. McCloghrie, M.T. Rose

RFC 1215
Convention for defining traps for use with the SNMP M. Rose

RFC 1227
SNMP MUX protocol and MIB M.T. Rose

RFC 1228
SNMP-DPI: Simple Network Management Protocol Distributed Program Interface G. Carpenter, B. Wijnen

RFC 1229
Extensions to the generic-interface MIB K. McCloghrie

RFC 1230
IEEE 802.4 Token Bus MIB K. McCloghrie, R. Fox

RFC 1231
IEEE 802.5 Token Ring MIB K. McCloghrie, R. Fox, E. Decker

RFC 1236
IP to X.121 address mapping for DDN L. Morales, P. Hasse

RFC 1256
ICMP Router Discovery Messages S. Deering, Ed.

RFC 1267
Border Gateway Protocol 3 (BGP-3) K. Lougheed, Y. Rekhter

RFC 1268
Application of the Border Gateway Protocol in the Internet Y. Rekhter, P. Gross

RFC 1269
Definitions of Managed Objects for the Border Gateway Protocol: Version 3 S. Willis, J. Burruss

RFC 1270
SNMP Communications Services F. Kastenholz, ed.

RFC 1285
FDDI Management Information Base J. Case

RFC 1315
Management Information Base for Frame Relay DTEs C. Brown, F. Baker, C. Carvalho

RFC 1321
The MD5 Message-Digest Algorithm R. Rivest

RFC 1323
TCP Extensions for High Performance V. Jacobson, R. Braden, D. Borman

RFC 1325
FYI on Questions and Answers: Answers to Commonly Asked "New Internet User" Questions G. Malkin, A. Marine

RFC 1327
Mapping between X.400 (1988)/ISO 10021 and RFC 822 S. Hardcastle-Kille

RFC 1340
Assigned Numbers J. Reynolds, J. Postel

RFC 1344
Implications of MIME for Internet Mail Gateways N. Bornstein

RFC 1349
Type of Service in the Internet Protocol Suite P. Almquist

RFC 1351
SNMP Administrative Model J. Davin, J. Galvin, K. McCloghrie

1370 z/OS Communications Server: IP Messages: Volume 2 (EZB, EZD)
RFC 1352
SNMP Security Protocols J. Galvin, K. McCloghrie, J. Davin

RFC 1353
Definitions of Managed Objects for Administration of SNMP Parties K. McCloghrie, J. Davin, J. Galvin

RFC 1354
IP Forwarding Table MIB F. Baker

RFC 1356
Multiprotocol Interconnect on X.25 and ISDN in the Packet Mode A. Malis, D. Robinson, R. Ullmann

RFC 1358
Charter of the Internet Architecture Board (IAB) L. Chapin

RFC 1363
A Proposed Flow Specification C. Partridge

RFC 1368
Definition of Managed Objects for IEEE 802.3 Repeater Devices D. McMaster, K. McCloghrie

RFC 1372
Telnet Remote Flow Control Option C. L. Hedrick, D. Borman

RFC 1374
IP and ARP on HIPPI J. Renwick, A. Nicholson

RFC 1381
SNMP MIB Extension for X.25 LAPB D. Throop, F. Baker

RFC 1382
SNMP MIB Extension for the X.25 Packet Layer D. Throop

RFC 1387
RIP Version 2 Protocol Analysis G. Malkin

RFC 1388
RIP Version 2 Carrying Additional Information G. Malkin

RFC 1389
RIP Version 2 MIB Extensions G. Malkin, F. Baker

RFC 1390
Transmission of IP and ARP over FDDI Networks D. Katz

RFC 1393
Traceroute Using an IP Option G. Malkin

RFC 1398
Definitions of Managed Objects for the Ethernet-Like Interface Types F. Kastenholz

RFC 1408
Telnet Environment Option D. Borman, Ed.

RFC 1413
Identification Protocol M. St. Johns

RFC 1416
Telnet Authentication Option D. Borman, ed.

RFC 1420
SNMP over IPX S. Bostock

RFC 1428
Transition of Internet Mail from Just-Send-8 to 8bit-SMTP/MIME G. Vaudreuil

RFC 1442

RFC 1443

Related protocol specifications 1371
 RFC 1445
Administrative Model for version 2 of the Simple Network Management Protocol (SNMPv2) J. Galvin, K. McCloghrie

 RFC 1447
Party MIB for version 2 of the Simple Network Management Protocol (SNMPv2) K. McCloghrie, J. Galvin

 RFC 1448

 RFC 1464
Using the Domain Name System to Store Arbitrary String Attributes R. Rosenbaum

 RFC 1469
IP Multicast over Token-Ring Local Area Networks T. Pusateri

 RFC 1483
Multiprotocol Encapsulation over ATM Adaptation Layer 5 Juha Heinanen

 RFC 1514
Host Resources MIB P. Grillo, S. Waldbusser

 RFC 1516
Definitions of Managed Objects for IEEE 802.3 Repeater Devices D. McMaster, K. McCloghrie

 RFC 1521
MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies N. Borenstein, N. Freed

 RFC 1535
A Security Problem and Proposed Correction With Widely Deployed DNS Software E. Gavron

 RFC 1536
Common DNS Implementation Errors and Suggested Fixes A. Kumar, J. Postel, C. Neuman, P. Danzig, S. Miller

 RFC 1537
Common DNS Data File Configuration Errors P. Beertema

 RFC 1540
Internet Official Protocol Standards J. Postel

 RFC 1571
Telnet Environment Option Interoperability Issues D. Borman

 RFC 1572
Telnet Environment Option S. Alexander

 RFC 1573
Evolution of the Interfaces Group of MIB-II K. McCloghrie, F. Kastenholz

 RFC 1577
Classical IP and ARP over ATM M. Laubach

 RFC 1583
OSPF Version 2 J. Moy

 RFC 1591
Domain Name System Structure and Delegation J. Postel

 RFC 1592

 RFC 1594
FYI on Questions and Answers—Answers to Commonly Asked "New Internet User" Questions A. Marine, J. Reynolds, G. Malkin

 RFC 1644
T/TCP—TCP Extensions for Transactions Functional Specification R. Braden
RFC 1646
TN3270 Extensions for LUname and Printer Selection C. Graves, T. Butts, M. Angel

RFC 1647
TN3270 Enhancements B. Kelly

RFC 1652
SMTP Service Extension for 8bit-MIMEtransport J. Klensin, N. Freed, M. Rose, E. Stefferud, D. Crocker

RFC 1664
Using the Internet DNS to Distribute RFC1327 Mail Address Mapping Tables C. Allochio, A. Bonito, B. Cole, S. Giordano, R. Hagens

RFC 1693
An Extension to TCP: Partial Order Service T. Connolly, P. Amer, P. Conrad

RFC 1695
Definitions of Managed Objects for ATM Management Version 8.0 using SMIv2 M. Ahmed, K. Tesink

RFC 1701
Generic Routing Encapsulation (GRE) S. Hanks, T. Li, D. Farinacci, P. Traina

RFC 1702
Generic Routing Encapsulation over IPv4 networks S. Hanks, T. Li, D. Farinacci, P. Traina

RFC 1706
DNS NSAP Resource Records B. Manning, R. Colella

RFC 1712
DNS Encoding of Geographical Location C. Farrell, M. Schulze, S. Pleitner D. Baldoni

RFC 1713
Tools for DNS debugging A. Romao

RFC 1723
RIP Version 2—Carrying Additional Information G. Malkin

RFC 1752
The Recommendation for the IP Next Generation Protocol S. Bradner, A. Mankin

RFC 1766
Tags for the Identification of Languages H. Alvestrand

RFC 1771
A Border Gateway Protocol 4 (BGP-4) Y. Rekhter, T. Li

RFC 1794
DNS Support for Load Balancing T. Brisco

RFC 1819

RFC 1826
IP Authentication Header R. Atkinson

RFC 1828
IP Authentication using Keyed MD5 P. Metzger, W. Simpson

RFC 1829
The ESP DES-CBC Transform P. Karn, P. Metzger, W. Simpson

RFC 1830
SMTP Service Extensions for Transmission of Large and Binary MIME Messages G. Vaudreuil

RFC 1831

RFC 1832
XDR: External Data Representation Standard R. Srinivasan

RFC 1833
Binding Protocols for ONC RPC Version 2 R. Srinivasan

Related protocol specifications 1373
RFC 1850
OSPF Version 2 Management Information Base F. Baker, R. Coltun

RFC 1854
SMTP Service Extension for Command Pipelining N. Freed

RFC 1869
SMTP Service Extensions J. Klensin, N. Freed, M. Rose, E. Stefferud, D. Crocker

RFC 1870
SMTP Service Extension for Message Size Declaration J. Klensin, N. Freed, K. Moore

RFC 1876
A Means for Expressing Location Information in the Domain Name System C. Davis, P. Vixie, T. Goodwin, I. Dickinson

RFC 1883
Internet Protocol, Version 6 (IPv6) Specification S. Deering, R. Hinden

RFC 1884
IP Version 6 Addressing Architecture R. Hinden, S. Deering, Eds.

RFC 1886
DNS Extensions to support IP version 6 S. Thomson, C. Huitema

RFC 1888
OSI NSAPs and IPv6 J. Bound, B. Carpenter, D. Harrington, J. Houldsworth, A. Lloyd

RFC 1891
SMTP Service Extension for Delivery Status Notifications K. Moore

RFC 1892
The Multipart/Report Content Type for the Reporting of Mail System Administrative Messages G. Vaudreuil

RFC 1894
An Extensible Message Format for Delivery Status NotificationsK. Moore, G. Vaudreuil

RFC 1901
Introduction to Community-based SNMPv2 J. Case, K. McCloghrie, M. Rose, S. Waldbusser

RFC 1902

RFC 1903

RFC 1904

RFC 1905

RFC 1906

RFC 1907

RFC 1908

RFC 1912
Common DNS Operational and Configuration Errors D. Barr
 RFC 1918  
Address Allocation for Private Internets Y. Rekhter, B. Moskowitz, D. Karrenberg, G.J. de Groot, E. Lear

 RFC 1928  

 RFC 1930  
Guidelines for creation, selection, and registration of an Autonomous System (AS) J. Hawkinson, T. Bates

 RFC 1939  
Post Office Protocol-Version 3 J. Myers, M. Rose

 RFC 1981  
Path MTU Discovery for IP version 6 J. McCann, S. Deering, J. Mogul

 RFC 1982  
Serial Number Arithmetic R. Elz, R. Bush

 RFC 1985  
SMTP Service Extension for Remote Message Queue Starting J. De Winter

 RFC 1995  
Incremental Zone Transfer in DNS M. Ohta

 RFC 1996  
A Mechanism for Prompt Notification of Zone Changes (DNS NOTIFY) P. Vixie

 RFC 2010  
Operational Criteria for Root Name Servers B. Manning, P. Vixie

 RFC 2011  
SNMPv2 Management Information Base for the Internet Protocol using SMIv2 K. McCloghrie, Ed.

 RFC 2012  

 RFC 2013  
SNMPv2 Management Information Base for the User Datagram Protocol using SMIv2 K. McCloghrie, Ed.

 RFC 2018  
TCP Selective Acknowledgement Options M. Mathis, J. Mahdavi, S. Floyd, A. Romanow

 RFC 2026  
The Internet Standards Process — Revision 3 S. Bradner

 RFC 2030  
Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI D. Mills

 RFC 2033  
Local Mail Transfer Protocol J. Myers

 RFC 2034  
SMTP Service Extension for Returning Enhanced Error Codes N. Freed

 RFC 2040  

 RFC 2045  
Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies N. Freed, N. Borenstein

 RFC 2052  
A DNS RR for specifying the location of services (DNS SRV) A. Gulbrandsen, P. Vixie

 RFC 2065  
Domain Name System Security Extensions D. Eastlake 3rd, C. Kaufman

 RFC 2066  
TELNET CHARSET Option R. Gellens

Related protocol specifications 1375
RFC 2080
RIPng for IPv6 G. Malkin, R. Minnear

RFC 2096
IP Forwarding Table MIB F. Baker

RFC 2104
HMAC: Keyed-Hashing for Message Authentication H. Krawczyk, M. Bellare, R. Canetti

RFC 2119
Keywords for use in RFCs to Indicate Requirement Levels S. Bradner

RFC 2133
Basic Socket Interface Extensions for IPv6 R. Gilligan, S. Thomson, J. Bound, W. Stevens

RFC 2136
Dynamic Updates in the Domain Name System (DNS UPDATE) P. Vixie, Ed., S. Thomson, Y. Rekhter, J. Bound

RFC 2137
Secure Domain Name System Dynamic Update D. Eastlake 3rd

RFC 2163
Using the Internet DNS to Distribute MIXER Conformant Global Address Mapping (MCGAM) C. Allocchio

RFC 2168
Resolution of Uniform Resource Identifiers using the Domain Name System R. Daniel, M. Mealling

RFC 2178
OSPF Version 2 J. Moy

RFC 2181
Clarifications to the DNS Specification R. Elz, R. Bush

RFC 2205

RFC 2210
The Use of RSVP with IETF Integrated Services J. Wroclawski

RFC 2211
Specification of the Controlled-Load Network Element Service J. Wroclawski

RFC 2212
Specification of Guaranteed Quality of Service S. Shenker, C. Partridge, R. Guerin

RFC 2215
General Characterization Parameters for Integrated Service Network Elements S. Shenker, J. Wroclawski

RFC 2217
Telnet Com Port Control Option G. Clarke

RFC 2219
Use of DNS Aliases for Network Services M. Hamilton, R. Wright

RFC 2228
FTP Security Extensions M. Horowitz, S. Lunt

RFC 2230
Key Exchange Delegation Record for the DNS R. Atkinson

RFC 2233
The Interfaces Group MIB using SMIv2 K. McCloghrie, F. Kastenholz

RFC 2240
A Legal Basis for Domain Name Allocation O. Vaughn

RFC 2246
The TLS Protocol Version 1.0 T. Dierks, C. Allen
RFC 2251

RFC 2253

RFC 2254
The String Representation of LDAP Search Filters T. Howes

RFC 2261
An Architecture for Describing SNMP Management Frameworks D. Harrington, R. Presuhn, B. Wijnen

RFC 2262

RFC 2271
An Architecture for Describing SNMP Management Frameworks D. Harrington, R. Presuhn, B. Wijnen

RFC 2273
SNMPv3 Applications D. Levi, P. Meyer, B. Stewart

RFC 2274

RFC 2275
View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP) B. Wijnen, R. Presuhn, K. McCloghrie

RFC 2279
UTF-8, a transformation format of ISO 10646 F. Yergeau

RFC 2292
Advanced Sockets API for IPv6 W. Stevens, M. Thomas

RFC 2308
Negative Caching of DNS Queries (DNS NCACHE) M. Andrews

RFC 2317
Classless IN-ADDR.ARPA delegation H. Eidnes, G. de Groot, P. Vixie

RFC 2320
Definitions of Managed Objects for Classical IP and ARP Over ATM Using SMIv2 (IPOA-MIB) M. Greene, J. Luciani, K. White, T. Kuo

RFC 2328
OSPF Version 2 J. Moy

RFC 2345
Domain Names and Company Name Retrieval J. Klensin, T. Wolf, G. Oglesby

RFC 2352
A Convention for Using Legal Names as Domain Names O. Vaughn

RFC 2355
TN3270 Enhancements B. Kelly

RFC 2358
Definitions of Managed Objects for the Ethernet-like Interface Types J. Flick, J. Johnson

RFC 2373
IP Version 6 Addressing Architecture R. Hinden, S. Deering

RFC 2374
An IPv6 Aggregatable Global Unicast Address Format R. Hinden, M. O'Dell, S. Deering

RFC 2375
IPv6 Multicast Address Assignments R. Hinden, S. Deering
RFC 2385
Protection of BGP Sessions via the TCP MD5 Signature Option A. Hefferman

RFC 2389
Feature negotiation mechanism for the File Transfer Protocol P. Hethmon, R. Elz

RFC 2401
Security Architecture for Internet Protocol S. Kent, R. Atkinson

RFC 2402
IP Authentication Header S. Kent, R. Atkinson

RFC 2403
The Use of HMAC-MD5–96 within ESP and AH C. Madson, R. Glenn

RFC 2404
The Use of HMAC-SHA–1–96 within ESP and AH C. Madson, R. Glenn

RFC 2405
The ESP DES-CBC Cipher Algorithm With Explicit IV C. Madson, N. Doraswamy

RFC 2406
IP Encapsulating Security Payload (ESP) S. Kent, R. Atkinson

RFC 2407
The Internet IP Security Domain of Interpretation for ISAKMPD. Piper

RFC 2408

RFC 2409
The Internet Key Exchange (IKE) D. Harkins, D. Carrel

RFC 2410
The NULL Encryption Algorithm and Its Use With IPsec R. Glenn, S. Kent,

RFC 2428
FTP Extensions for IPv6 and NATs M. Allman, S. Ostermann, C. Metz

RFC 2445
Internet Calendaring and Scheduling Core Object Specification (iCalendar) F. Dawson, D. Stenerson

RFC 2459
Internet X.509 Public Key Infrastructure Certificate and CRL Profile R. Housley, W. Ford, W. Polk, D. Solo

RFC 2460
Internet Protocol, Version 6 (IPv6) Specification S. Deering, R. Hinden

RFC 2461

RFC 2462
IPv6 Stateless Address Autoconfiguration S. Thomson, T. Narten

RFC 2463
Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification A. Conta, S. Deering

RFC 2464
Transmission of IPv6 Packets over Ethernet Networks M. Crawford

RFC 2466

RFC 2476
Message Submission R. Gellens, J. Klensin

RFC 2487
SMTP Service Extension for Secure SMTP over TLS P. Hoffman

RFC 2505
Anti-Spam Recommendations for SMTP MTAs G. Lindberg
RFC 2523
Photuris: Extended Schemes and Attributes P. Karn, W. Simpson

RFC 2535
Domain Name System Security Extensions D. Eastlake 3rd

RFC 2538
Storing Certificates in the Domain Name System (DNS) D. Eastlake 3rd, O. Gudmundsson

RFC 2539
Storage of Diffie-Hellman Keys in the Domain Name System (DNS) D. Eastlake 3rd

RFC 2540
Detached Domain Name System (DNS) Information D. Eastlake 3rd

RFC 2554
SMTP Service Extension for Authentication J. Myers

RFC 2570
Introduction to Version 3 of the Internet-standard Network Management Framework J. Case, R. Mundy, D. Partain, B. Stewart

RFC 2571
An Architecture for Describing SNMP Management Frameworks B. Wijnen, D. Harrington, R. Presuhn

RFC 2572

RFC 2573
SNMP Applications D. Levi, P. Meyer, B. Stewart

RFC 2574

RFC 2575
View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP) B. Wijnen, R. Presuhn, K. McCloghrie

RFC 2576

RFC 2578

RFC 2579
Textual Conventions for SMIv2 K. McCloghrie, D. Perkins, J. Schoenwaelder

RFC 2580
Conformance Statements for SMIv2 K. McCloghrie, D. Perkins, J. Schoenwaelder

RFC 2581
TCP Congestion Control M. Allman, V. Paxson, W. Stevens

RFC 2583
Guidelines for Next Hop Client (NHC) Developers R. Carlson, L. Winkler

RFC 2591
Definitions of Managed Objects for Scheduling Management Operations D. Levi, J. Schoenwaelder

RFC 2625
IP and ARP over Fibre Channel M. Rajagopal, R. Bhagwat, W. Rickard

RFC 2635
Don’t SPEW A Set of Guidelines for Mass Unsolicited Mailings and Postings (spam*) S. Hambridge, A. Lunde

RFC 2637

Related protocol specifications 1379
RFC 2953
Telnet Encryption: DES 64 bit Output Feedback T. Ts'o

RFC 2992
Analysis of an Equal-Cost Multi-Path Algorithm C. Hopps

RFC 3019
IP Version 6 Management Information Base for The Multicast Listener Discovery Protocol B. Haberman, R. Worzella

RFC 3060
Policy Core Information Model—Version 1 Specification B. Moore, E. Ellesson, J. Strassner, A. Westerinen

RFC 3152
Delegation of IP6.ARPA R. Bush

RFC 3164
The BSD Syslog Protocol C. Lonvick

RFC 3207
SMTP Service Extension for Secure SMTP over Transport Layer Security P. Hoffman

RFC 3226
DNSSEC and IPv6 A6 aware server/resolver message size requirements O. Gudmundsson

RFC 3291
Textual Conventions for Internet Network Addresses M. Daniele, B. Haberman, S. Routhier, J. Schoenwaelder

RFC 3363

RFC 3376
Internet Group Management Protocol, Version 3 B. Cain, S. Deering, I. Kouvelas, B. Fenner, A. Thyagarajan

RFC 3390
Increasing TCP's Initial Window M. Allman, S. Floyd, C. Partridge

RFC 3410
Introduction and Applicability Statements for Internet-Standard Management Framework J. Case, R. Mundy, D. Partain, B. Stewart

RFC 3411

RFC 3412

RFC 3413

RFC 3414

RFC 3415
View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP) B. Wijnen, R. Presuhn, K. McCloghrie

RFC 3416

Related protocol specifications 1381
RFC 4022
Management Information Base for the Transmission Control Protocol (TCP) R. Raghunarayan

RFC 4106
The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP) J. Viega, D. McGrew

RFC 4109
Algorithms for Internet Key Exchange version 1 (IKEv1) P. Hoffman

RFC 4113
Management Information Base for the User Datagram Protocol (UDP) B. Fenner, J. Flick

RFC 4191
Default Router Preferences and More-Specific Routes R. Draves, D. Thaler

RFC 4217
Securing FTP with TLS P. Ford-Hutchinson

RFC 4292
IP Forwarding Table MIB B. Haberman

RFC 4293
Management Information Base for the Internet Protocol (IP) S. Routhier

RFC 4301
Security Architecture for the Internet Protocol S. Kent, K. Seo

RFC 4302
IP Authentication Header S. Kent

RFC 4303
IP Encapsulating Security Payload (ESP) S. Kent

RFC 4304
Extended Sequence Number (ESN) Addendum to IPsec Domain of Interpretation (DOI) for Internet Security Association and Key Management Protocol (ISAKMP) S. Kent

RFC 4307
Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2) J. Schiller

RFC 4308
Cryptographic Suites for IPsec P. Hoffman

RFC 4434
The AES-XCBC-PRF-128 Algorithm for the Internet Key Exchange Protocol P. Hoffman

RFC 4443
Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification A. Conta, S. Deering

RFC 4552
Authentication/Confidentiality for OSPFv3 M. Gupta, N. Melam

RFC 4678
Server/Application State Protocol v1 A. Bivens

RFC 4753
ECP Groups for IKE and IKEv2 D. Fu, J. Solinas

RFC 4754
IKE and IKEv2 Authentication Using the Elliptic Curve Digital Signature Algorithm (ECDSA) D. Fu, J. Solinas

RFC 4809

RFC 4835
Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH) V. Manral
RFC 4862
IPv6 Stateless Address Autoconfiguration S. Thomson, T. Narten, T. Jinmei

RFC 4868
Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec S. Kelly, S. Frankel

RFC 4869
Suite B Cryptographic Suites for IPsec L. Law, J. Solinas

RFC 4941
Privacy Extensions for Stateless Address Autoconfiguration in IPv6 T. Narten, R. Draves, S. Krishnan

RFC 4945
The Internet IP Security PKI Profile of IKEv1/ISAKMP, IKEv2, and PKIX B. Korver

RFC 5014
IPv6 Socket API for Source Address Selection E. Nordmark, S. Chakrabarti, J. Laganier

RFC 5095
Deprecation of Type 0 Routing Headers in IPv6 J. Abley, P. Savola, G. Neville-Neil

RFC 5175
IPv6 Router Advertisement Flags Option B. Haberman, Ed., R. Hinden

RFC 5282
Using Authenticated Encryption Algorithms with the Encrypted Payload of the Internet Key Exchange version 2 (IKEv2) Protocol D. Black, D. McGrew

RFC 5996
Internet Key Exchange Protocol Version 2 (IKEv2) C. Kaufman, P. Hoffman, Y. Nir, P. Eronen

Internet drafts
Internet drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Other groups can also distribute working documents as Internet drafts. You can see Internet drafts at http://www.ietf.org/ID.html.
Appendix B. Accessibility

Publications for this product are offered in Adobe Portable Document Format (PDF) and should be compliant with accessibility standards. If you experience difficulties when using PDF files, you can view the information through the z/OS Internet Library website http://www.ibm.com/systems/z/os/zos/library/bkserv/ or IBM Knowledge Center http://www.ibm.com/support/knowledgecenter/. If you continue to experience problems, send a message to Contact z/OS web page (www.ibm.com/systems/z/os/zos/webqs.html) or write to:

IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/OS enable users to:

- Use assistive technologies such as screen readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard
- Customize display attributes such as color, contrast, and font size

Using assistive technologies

Assistive technology products, such as screen readers, function with the user interfaces found in z/OS. Consult the assistive technology documentation for specific information when using such products to access z/OS interfaces.

Keyboard navigation of the user interface

Users can access z/OS user interfaces using TSO/E or ISPF. See z/OS TSO/E Primer, z/OS TSO/E User's Guide, and z/OS ISPF User's Guide Vol I for information about accessing TSO/E and ISPF interfaces. These guides describe how to use TSO/E and ISPF, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.
Notices

This information was developed for products and services that are offered in the USA or elsewhere.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user’s responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation North Castle Drive, MD-NC119 Armonk, NY 10504-1785 United States of America

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan Ltd. 19-21, Nihonbashi-Hakozakicho, Chuo-ku Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

This information could include missing, incorrect, or broken hyperlinks. Hyperlinks are maintained in only the HTML plug-in output for the Knowledge Centers. Use of hyperlinks in other output formats of this information is at your own risk.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation Site Counsel 2455 South Road Poughkeepsie, NY 12601-5400 USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.
Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.
IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user’s name, email address, phone number, or other personally identifiable information for purposes of enhanced user usability and single sign-on configuration. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM’s Privacy Policy at ibm.com/privacy and IBM’s Online Privacy Statement at ibm.com/privacy/details in the section entitled “Cookies, Web Beacons and Other Technologies,” and the “IBM Software Products and Software-as-a-Service Privacy Statement” at ibm.com/software/info/product-privacy.

Policy for unsupported hardware

Various z/OS elements, such as DFSMS, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: IBM Lifecycle Support for z/OS (www.ibm.com/software/support/systemsz/lifecycle)
- For information about currently-supported IBM hardware, contact your IBM representative.
Policy for unsupported hardware

Various z/OS elements, such as DFSMS, HCD, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.
This bibliography contains descriptions of the documents in the z/OS Communications Server library. z/OS Communications Server documentation is available online at the z/OS Internet Library web page at http://www.ibm.com/systems/z/os/zos/library/bkserv/.

### z/OS Communications Server library updates

Updates to documents are also available on RETAIN and in information APARs (info APARs). Go to http://www.software.ibm.com/support to view information APARs.

- z/OS V2R1 Communications Server New Function APAR Summary
- z/OS V2R2 Communications Server New Function APAR Summary
- z/OS V2R3 Communications Server New Function APAR Summary

### z/OS Communications Server information

z/OS Communications Server product information is grouped by task in the following tables.

#### Planning

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: New Function Summary</td>
<td>GC27-3664</td>
<td>This document is intended to help you plan for new IP or SNA functions, whether you are migrating from a previous version or installing z/OS for the first time. It summarizes what is new in the release and identifies the suggested and required modifications needed to use the enhanced functions.</td>
</tr>
<tr>
<td>z/OS Communications Server: IPv6 Network and Application Design Guide</td>
<td>SC27-3663</td>
<td>This document is a high-level introduction to IPv6. It describes concepts of z/OS Communications Server's support of IPv6, coexistence with IPv4, and migration issues.</td>
</tr>
</tbody>
</table>

#### Resource definition, configuration, and tuning

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: IP Configuration Guide</td>
<td>SC27-3650</td>
<td>This document describes the major concepts involved in understanding and configuring an IP network. Familiarity with the z/OS operating system, IP protocols, z/OS UNIX System Services, and IBM Time Sharing Option (TSO) is recommended. Use this document with the z/OS Communications Server: IP Configuration Reference.</td>
</tr>
</tbody>
</table>
**z/OS Communications Server: IP Configuration Reference**

SC27-3651

This document presents information for people who want to administer and maintain IP. Use this document with the z/OS Communications Server: IP Configuration Guide. The information in this document includes:

- TCP/IP configuration data sets
- Configuration statements
- Translation tables
- Protocol number and port assignments

**z/OS Communications Server: SNA Network Implementation Guide**

SC27-3672

This document presents the major concepts involved in implementing an SNA network. Use this document with the z/OS Communications Server: SNA Resource Definition Reference.

**z/OS Communications Server: SNA Resource Definition Reference**

SC27-3675

This document describes each SNA definition statement, start option, and macroinstruction for user tables. It also describes NCP definition statements that affect SNA. Use this document with the z/OS Communications Server: SNA Network Implementation Guide.

**z/OS Communications Server: SNA Resource Definition Samples**

SC27-3676

This document contains sample definitions to help you implement SNA functions in your networks, and includes sample major node definitions.

**z/OS Communications Server: IP Network Print Facility**

SC27-3658

This document is for systems programmers and network administrators who need to prepare their network to route SNA, JES2, or JES3 printer output to remote printers using TCP/IP Services.

---

**Operation**

**z/OS Communications Server: IP User's Guide and Commands**

SC27-3662

This document describes how to use TCP/IP applications. It contains requests with which a user can log on to a remote host using Telnet, transfer data sets using FTP, send electronic mail, print on remote printers, and authenticate network users.

**z/OS Communications Server: IP System Administrator's Commands**

SC27-3661

This document describes the functions and commands helpful in configuring or monitoring your system. It contains system administrator’s commands, such as TSO NETSTAT, PING, TRACERTE and their UNIX counterparts. It also includes TSO and MVS commands commonly used during the IP configuration process.

**z/OS Communications Server: SNA Operation**

SC27-3673

This document serves as a reference for programmers and operators requiring detailed information about specific operator commands.

**z/OS Communications Server: Quick Reference**

SC27-3665

This document contains essential information about SNA and IP commands.
### Customization

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
</table>
| z/OS Communications Server: SNA Customization | SC27-3666 | This document enables you to customize SNA, and includes the following information:  
  - Communication network management (CNM) routing table  
  - Logon-interpret routine requirements  
  - Logon manager installation-wide exit routine for the CLU search exit  
  - TSO/SNA installation-wide exit routines  
  - SNA installation-wide exit routines |

### Writing application programs

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference</td>
<td>SC27-3660</td>
<td>This document describes the syntax and semantics of program source code necessary to write your own application programming interface (API) into TCP/IP. You can use this interface as the communication base for writing your own client or server application. You can also use this document to adapt your existing applications to communicate with each other using sockets over TCP/IP.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP CICS Sockets Guide</td>
<td>SC27-3649</td>
<td>This document is for programmers who want to set up, write application programs for, and diagnose problems with the socket interface for CICS® using z/OS TCP/IP.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP IMS Sockets Guide</td>
<td>SC27-3653</td>
<td>This document is for programmers who want application programs that use the IMS TCP/IP application development services provided by the TCP/IP Services of IBM.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP Programmer's Guide and Reference</td>
<td>SC27-3659</td>
<td>This document describes the syntax and semantics of a set of high-level application functions that you can use to program your own applications in a TCP/IP environment. These functions provide support for application facilities, such as user authentication, distributed databases, distributed processing, network management, and device sharing. Familiarity with the z/OS operating system, TCP/IP protocols, and IBM Time Sharing Option (TSO) is recommended.</td>
</tr>
<tr>
<td>z/OS Communications Server: SNA Programming</td>
<td>SC27-3674</td>
<td>This document describes how to use SNA macroinstructions to send data to and receive data from (1) a terminal in either the same or a different domain, or (2) another application program in either the same or a different domain.</td>
</tr>
<tr>
<td>z/OS Communications Server: SNA Programmer’s LU 6.2 Guide</td>
<td>SC27-3669</td>
<td>This document describes how to use the SNA LU 6.2 application programming interface for host application programs. This document applies to programs that use only LU 6.2 sessions or that use LU 6.2 sessions along with other session types. (Only LU 6.2 sessions are covered in this document.)</td>
</tr>
</tbody>
</table>
### Diagnosis

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures and z/OS Communications Server: SNA Diagnosis Vol 2, FFST Dumps and the VIT</td>
<td>GC27-3667, GC27-3668</td>
<td>These documents help you identify an SNA problem, classify it, and collect information about it before you call the IBM Support Center. The information collected includes traces, dumps, and other problem documentation.</td>
</tr>
<tr>
<td>z/OS Communications Server: SNA Data Areas Volume 1 and z/OS Communications Server: SNA Data Areas Volume 2</td>
<td>GC31-6852, GC31-6853</td>
<td>These documents describe SNA data areas and can be used to read an SNA dump. They are intended for IBM programming service representatives and customer personnel who are diagnosing problems with SNA.</td>
</tr>
<tr>
<td>z/OS Communications Server: ACF/TAP Trace Analysis Handbook</td>
<td>GC27-3645</td>
<td>This document explains how to gather the trace data that is collected and stored in the host processor. It also explains how to use the Advanced Communications Function/Trace Analysis Program (ACF/TAP) service aid to produce reports for analyzing the trace data information.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP Diagnosis Guide</td>
<td>GC27-3652</td>
<td>This document explains how to diagnose TCP/IP problems and how to determine whether a specific problem is in the TCP/IP product code. It explains how to gather information for and describe problems to the IBM Software Support Center.</td>
</tr>
</tbody>
</table>

### Messages and codes

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
</table>
| z/OS Communications Server: SNA Messages                           | SC27-3671   | This document describes the ELM, IKT, IST, IUT, IVT, and USS messages. Other information in this document includes:  
  - Command and RU types in SNA messages  
  - Node and ID types in SNA messages  
  - Supplemental message-related information |
<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: IP Messages Volume 1 (EZA)</td>
<td>SC27-3654</td>
<td>This volume contains TCP/IP messages beginning with EZA.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP Messages Volume 2 (EZB, EZD)</td>
<td>SC27-3655</td>
<td>This volume contains TCP/IP messages beginning with EZB or EZD.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP Messages Volume 3 (EZY)</td>
<td>SC27-3656</td>
<td>This volume contains TCP/IP messages beginning with EZY.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP Messages Volume 4 (EZZ, SNM)</td>
<td>SC27-3657</td>
<td>This volume contains TCP/IP messages beginning with EZZ and SNM.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP and SNA Codes</td>
<td>SC27-3648</td>
<td>This document describes codes and other information that appear in z/OS Communications Server messages.</td>
</tr>
</tbody>
</table>
Communicating your comments to IBM

If you especially like or dislike anything about this document, you can send us comments electronically by using one of the following methods:

**Internet email:**
comsvrcf@us.ibm.com

**World Wide Web:**

If you would like a reply, be sure to include your name, address, and telephone number. Make sure to include the following information in your comment or note:

- Title and order number of this document
- Page number or topic related to your comment

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this document. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.