z/OS Communications Server
Version 2 Release 3

IP Messages:
Volume 3 (EZY)

IBM
Note:
Before using this information and the product it supports, be sure to read the general information under “Notices” on page 1077.
Contents

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

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

Summary of changes....................................................................................................... xv

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

Chapter 2. EZY0xxxx messages................................................................................... 7

Chapter 3. EZY1xxxx messages.................................................................................. 121

Chapter 4. EZY2xxxx messages.................................................................................. 263

Chapter 5. EZY3xxxx messages.................................................................................. 351

Chapter 6. EZY4xxxx messages.................................................................................. 401

Chapter 7. EZY5xxxx messages.................................................................................. 503

Chapter 8. EZY6xxxx messages.................................................................................. 505

Chapter 9. EZYFxxxx messages.................................................................................. 575

Chapter 10. EZYPxxxx messages............................................................................... 657

Chapter 11. EZYRxxxx messages............................................................................... 703

Chapter 12. EZYTxxxx messages............................................................................... 767

Chapter 13. EZYXxxxx messages............................................................................... 845

Appendix A. Related protocol specifications............................................................... 1055

Appendix B. Accessibility............................................................................................. 1075

Notices............................................................................................................................ 1077

Bibliography.................................................................................................................... 1081

Communicating your comments to IBM................................................................. 1087
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 within 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 Programmer’s 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, “EZY0xxxx messages,” on page 7 contains messages in the EZY0xxxx range.
- Chapter 3, “EZY1xxxx messages,” on page 121 contains messages in the EZY1xxxx range.
- Chapter 4, “EZY2xxxx messages,” on page 263 contains messages in the EZY2xxxx range.
- Chapter 5, “EZY3xxxx messages,” on page 351 contains messages in the EZY3xxxx range.
- Chapter 6, “EZY4xxxx messages,” on page 401 contains messages in the EZY4xxxx range.
- Chapter 7, “EZY5xxxx messages,” on page 503 contains messages in the EZY5xxxx range.
- Chapter 8, “EZY6xxxx messages,” on page 505 contains messages in the EZY6xxxx range.
- Chapter 9, “EZYFxxxx messages,” on page 575 contains messages in the EZYFxxxx range.
- Chapter 10, “EZYPxxxx messages,” on page 657 contains messages in the EZYPxxxx range.
- Chapter 11, “EZYRxxxx messages,” on page 703 contains messages in the EZYRxxxx range.
- Chapter 12, “EZYTxxxx messages,” on page 767 contains messages in the EZYTxxxx range.
- Chapter 13, “EZYXxxxx messages,” on page 845 contains messages in the EzyXxxxx range.
- Appendix A, “Related protocol specifications,” on page 1055 lists the related protocol specifications for TCP/IP.
- Appendix B, “Accessibility,” on page 1075 describes accessibility features to help users with physical disabilities.
- “Notices” on page 1077 contains notices and trademarks used in this document.
About this document

• “Bibliography” on page 1081 contains descriptions of the documents in the z/OS Communications Server library.

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 1087.

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 1081, 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

http://www.software.ibm.com/network/commsserver/

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

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 3 (EZY), which supported z/OS Version 2 Release 2.

December 2018
Changed information

• EZYFS57I

January 2018
Changed information

• EZYFS57I

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

This information contains no technical change for this release.

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:
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:

<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>
3 **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.

4 **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.

5 **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.

6 **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.

7 **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.

8 **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.

9 **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.

10 **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.

11 **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.

12 **Emulators:** This message is issued by an emulator program.

13 Reserved for customer use.

14 Reserved for customer use.

15 Reserved for customer use.

16 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>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
System Failure: This message indicates that an error that cannot be corrected occurs. To continue, the operator must restart the system.

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.

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

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

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.

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

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

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.

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

This message is issued in response to a TRACK command.

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.

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

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.

<table>
<thead>
<tr>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZZ8453I jobtype STORAGE</td>
<td></td>
</tr>
<tr>
<td>EZZ8454I jobname STORAGE CURRENT MAXIMUM LIMIT</td>
<td></td>
</tr>
<tr>
<td>EZD2818I location</td>
<td></td>
</tr>
<tr>
<td>EZZ845S1 strogagetype current maximum limit</td>
<td></td>
</tr>
<tr>
<td>EZZ8459I DISPLAY TCPIP STOR COMPLETED SUCCESSFULLY</td>
<td></td>
</tr>
</tbody>
</table>

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. EZY0xxxx messages

EZY0002I   LINKNAME: linkname TRACE: ON/OFF

Explanation
This is the first message displayed when the PKTTRACE LIST command is issued. The link name is displayed, as well as the current trace level. If trace is ON, then the trace options listed are currently set and active. If trace is OFF, then the trace options displayed are currently set, but no tracing will occur for the link.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PKTLIST

EZY0003I   Current Trace Options:

Explanation
This message precedes the display of current trace options when the PKTTRACE LIST command is issued.

System action
Processing continues. The trace options and trace level for the link are unchanged.

Operator response
None.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PKTLIST

EZY0004I   Protocol - ALL
**Explanation**
All protocols will be traced for IP packets passing through the link associated with this message. This message can be displayed when the PKTTRACE LIST command is issued.

**System action**
Processing continues. The trace options and trace level for the link are unchanged.

**Operator response**
None.

**System programmer response**
None.

**Module**
IUPKTTRC

**Procedure name**
PKTLIST

**EZY0005I Protocol - protocol**

**Explanation**
A particular well-known protocol, ICMP, TCP, or UDP, will be traced for IP packets passing through the link associated with this message. This message can be displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
IUPKTTRC

**Procedure name**
PKTLIST

**EZY0006I Protocol - protocol_number**

**Explanation**
A particular protocol number will be traced for IP packets passing through the link associated with this message. This message can be displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.
System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
IUPKTTTRC

Procedure name
PKTLIST

EZY0007I  Src Port - ALL

Explanation
All source ports will be traced for TCP or UDP IP packets passing through the link associated with this message. This message can be displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
IUPKTTTRC

Procedure name
PKTLIST

EZY0008I  Src Port - source port

Explanation
A particular source port number will be traced for TCP or UDP IP packets passing through the link associated with this message. Packets that are not TCP or UDP will not be traced. This message can be displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

System action
Processing continues.
Operator response
None.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PKTLIST

EZY0009I  Dest Port - ALL

Explanation
All destination ports will be traced for TCP or UDP IP packets passing through the link associated with this message. This message is displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PKTLIST

EZY0010I  Dest Port - destination port

Explanation
A particular destination port number will be traced for TCP or UDP IP packets passing through the link associated with this message. Packets that are not TCP or UDP IP will not be traced. This message is displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

System action
Processing continues.

Operator response
None.
System programmer response
None.

Module
IUPKTTRC

Procedure name
PKTLIST

**EZY0011I** IP address - octet1.octet2.octet3.octet4

Explanation
This message displays the IP address. This can be used in conjunction with the subnet address mask to determine if IP packets passing through the link associated with this message will be traced. This message is displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PKTLIST

**EZY0012I** Abbrev Len - length

Explanation
Abbreviated IP packets will be written to the trace storage device for IP packets that are traced on the link associated with this message. The length field indicates how many bytes of the IP packet will be written, starting from the beginning of the IP header. For an ABBREV length of 0, only the packet trace header will be written. This message is displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

System action
Processing continues.

Operator response
None.

System programmer response
None.
EZY0013I  Abbrev Len - FULL

Explanation
Complete IP packets will be written to the trace storage device for IP packets that are traced on the link associated with this message. This message is displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

System action
Processing continues.

Operator response
None.

System programmer response
None.

EZY0014I  Subnet mask - octet1.octet2.octet3.octet4

Explanation
The subnet address mask that will be used with the IP address to determine if IP packets passing through the link associated with this message will be traced. This message is displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

System action
Processing continues.

Operator response
None.

System programmer response
None.
PKTLIST

EZY0015I  IP address - ALL

Explanation
All IP addresses will be traced for IP packets passing through the link associated with this message. This message is displayed when the PKTTRACE LIST command is issued. The trace options and trace level for the link are unchanged.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PKTLIST

EZY0016I  Clearing pkttrace options for linkname linkname

Explanation
The PKTTRACE CLEAR command has been issued for the link or links identified. Trace options for the link or links are set to default values and tracing is turned off.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
IUPKTTRC

Procedure name
DOPKTTRC

EZY0017I  PKTTRACE command accepted.
**Explanation**
The PKTTRACE command has been parsed without error. Trace options specified in the PKTTRACE command have been set. This message is displayed when a PKTTRACE command has been issued.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
IUPKTTRC

**Procedure name**
PARSEPKT

**EZY0022W  Trace options with LIST/CLEAR ignored**

**Explanation**
Trace options other than LINKNAME were specified in a PKTTRACE LIST or CLEAR command. The LIST or CLEAR command is executed; the other trace options are ignored.

**System action**
Processing continues.

**Operator response**
Reenter the PKTTRACE LIST or CLEAR command, specifying only the LINKNAME option, if required.

**System programmer response**
None.

**Module**
IUPKTTRC

**Procedure name**
PARSEPKT

**EZY0023W  SRCPORT/DESTPORT specified, but protocol not UDP/TCP**

**Explanation**
The SRCPORT or DESTPORT trace option was specified in a PKTTRACE command, but the PROTOCOL trace option specified was not UDP or TCP. No IP packets will be traced for the specified link or links. The trace options are set as specified on the PKTTRACE command.
System action
Processing continues.

Operator response
Reenter the PKTTRACE command, specifying a protocol of UDP or TCP.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PARSEPKT

EZY0024W Host identified by IP address ignored, SUBNET mask has been supplied.

Explanation
The value provided with the SUBNET parameter is used to distinguish the network and host portions of an IP address. A SUBNET mask is used to trace all packets from a particular network, it is therefore not valid to set host bits in conjunction with the SUBNET parameter. This message is generated if an IP address is supplied that has a nonzero host address. The trace options are set as specified on the PKTTRACE command. The host component of the IP address is ignored.

System action
Processing continues.

Operator response
Reenter the PKTTRACE command, ensuring all the host address bits are specified as zero in the IP address supplied with the IP trace option.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PARSEPKT

EZY0025W No IP address specified with SUBNET

Explanation
The SUBNET trace option was specified in a PKTTRACE command, but no IP trace option was specified. The SUBNET and IP options are used in combination to determine if IP packets are to be traced. If no IP option is specified, then the default IP address mask, or that specified by a previous PKTTRACE command, will be used. The trace options are set as specified on the PKTTRACE command.
System action
Processing continues.

Operator response
Reenter the PKTTRACE command, specifying a valid IP address option with the SUBNET option.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PARSEPKT

EZY0026W  No options specified, no changes applied

Explanation
No options were specified with the PKTTRACE command. No changes will be made to the trace options or trace level for any links.

System action
Processing continues.

Operator response
None. Use this command to re-enable packet trace for the driver, when GTRACE has failed writing to the trace storage device. Enable or re-enable GTRACE, and then enter the PKTTRACE command with no options. Any PKTTRACE command, including one with options, could be used in this situation.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PARSEPKT

EZY0032E  Invalid option option.

Explanation
The specified text is not a valid option for the PKTTRACE command. The command is ignored. No changes are made to any trace options.

System action
Processing continues.
Operator response
Reenter the PKTTRACE command, specifying a valid option. The valid options are:

- ON
- OFF
- LIST
- CLEAR
- LINKNAME=
- PROT=
- IP=
- SUBNET=
- SRCPORT=
- DESTPORT=
- FULL
- ABBREV
- ABBREV=

System programmer response
None.

Module
IUPKTTRC

Procedure name
PARSEPKT

EZY0033E  Option option duplicated

Explanation
Any PKTTRACE option can be specified only once in the same PKTTRACE command. The command is ignored. No changes are made to any trace options.

System action
Processing continues.

Operator response
Reenter the PKTTRACE command, specifying only one of each required option.

System programmer response
None.

Module
IUPKTTRC

Procedure name
PARSEPKT

EZY0034E  Specify one only of ON/OFF/LIST/CLEAR
**Explanation**

Only one of the PKTTRACE options ON, OFF, LIST, or CLEAR can be specified in the same PKTTRACE command. The command is ignored. No changes are made to any trace options.

**System action**

Processing continues.

**Operator response**

Reenter the PKTTRACE command, specifying only one of the options ON, OFF, LIST, or CLEAR.

**System programmer response**

None.

**Module**

IUPKTTRC

**Procedure name**

PARSEPKT

**EZY0035E**  Specify one only of FULL/ABBREV

**Explanation**

Only one of the PKTTRACE options FULL or ABBREV can be specified in the same PKTTRACE command. The command is ignored. No changes are made to any trace options.

**System action**

Processing continues.

**Operator response**

Reenter the PKTTRACE command, specifying either the FULL or ABBREV option.

**System programmer response**

None.

**Module**

IUPKTTRC

**Procedure name**

PARSEPKT

**EZY0036E**  Invalid character specified after keyword option

**Explanation**

A required option value delimiter has not been specified. The following PKTTRACE options all require an equal sign delimiter:

- LINKNAME=
- PROT=
- IP=
• SUBNET=
• SRCPORT=
• DESTPORT=
• ABBREV=

The command is ignored. No changes are made to any trace options.

**System action**
Processing continues.

**Operator response**
Reenter the PKTTRACE command, specifying the option correctly.

**System programmer response**
None.

**Module**
IUPKTTRC

**Procedure name**
PARSEPKT

---

**EZY0037E  Invalid IP address address**

**Explanation**
The supplied IP address is not valid. The address must be specified in dotted decimal notation, with exactly four octets, for example, 193.9.34.8, or the address can be specified as a single asterisk (*), to denote all IP addresses. The command is ignored. No changes are made to any trace options.

**System action**
Processing continues.

**Operator response**
Reenter the PKTTRACE command, specifying a valid IP address.

**System programmer response**
None.

**Module**
IUPKTTRC

**Procedure name**
PARSEPKT

---

**EZY0038E  No value after '=' for keyword option**

**Explanation**
A required option value has not been specified. The following PKTTRACE options all require a value:
The command is ignored. No changes are made to any trace options.

**System action**
Processing continues.

**Operator response**
Reenter the PKTTRACE command, specifying the required value.

**Module**
IUPKTTRC

**Procedure name**
PARSEPKT

**EZY0039E Undefined keyword only option option**

**Explanation**
The specified text is not a valid keyword-only option for the PKTTRACE command. A keyword-only option has either no value, or a default value. Any single word delimited by spaces is considered a keyword-only option because keyword value options are immediately followed by an equals sign. The command is ignored. No changes are made to any trace options.

**System action**
Processing continues.

**Operator response**
Reenter the PKTTRACE command, specifying a valid option. The valid keyword-only options are:
- ON
- OFF
- LIST
- CLEAR
- FULL
- ABBREV

**Module**
IUPKTTRC

**Procedure name**
PARSEPKT
EZY0040E  Invalid protocol protocol

Explanation
The supplied protocol value is not valid. The command is ignored. No changes are made to any trace options.

System action
Processing continues.

Operator response
Reenter the PKTTRACE command, specifying a valid protocol value. Protocol values must be one of the following:

- A number from 0 to 255
- One of the labels ICMP, TCP, or UDP
- A single asterisk (*) used to denote all protocols

System programmer response
None.

Module
IUPKTRC

Procedure name
PARSEPKT

EZY0041E  Invalid subnet address address

Explanation
The supplied subnet address is not valid. The command is ignored. No changes are made to any trace options.

System action
Processing continues.

Operator response
Reenter the PKTTRACE command, specifying a valid subnet address. The address must be specified in dotted decimal notation, with exactly 4 octets, for example, 193.9.34.8.

System programmer response
None.

Module
IUPKTRC

Procedure name
PARSEPKT

EZY0042E  Invalid abbrev value abbrev_len
**Explanation**
The supplied ABBREV value is not valid. The command is ignored. No changes are made to any trace options.

**System action**
Processing continues.

**Operator response**
Reenter the PKTTRACE command, specifying a valid ABBREV value. Valid values are integers in the range 0 to 65 535.

**System programmer response**
None.

**Module**
IUPKTTRC

**Procedure name**
PARSEPKT

---

**EZY0043E**  Invalid source port *port*

**Explanation**
The supplied source port is not valid. The command is ignored. No changes are made to any trace options.

**System action**
Processing continues.

**Operator response**
Reenter the PKTTRACE command, specifying a valid source port value. Valid values are integers in the range 0 to 65 535, or a single asterisk (*) used to indicate all source ports.

**System programmer response**
None.

**Module**
IUPKTTRC

**Procedure name**
PARSEPKT

---

**EZY0044E**  Invalid dest port *port*

**Explanation**
The supplied destination port is not valid. The command is ignored. No changes are made to any trace options.

**System action**
Processing continues.
Operator response
Reenter the PKTTRACE command, specifying a valid destination port value. Valid values are integers in the range 0 to 65 535, or a single asterisk (*) used to indicate all destination ports.

System programmer response
None.

Module
IUPKTRC

Procedure name
PARSEPKT

EZY0045E  Invalid linkname linkname

Explanation
The LINKNAME value is too long. Valid link names must be no more than 8 characters. The command is ignored. No changes are made to any trace options.

System action
Processing continues.

Operator response
Reenter the PKTTRACE command, specifying a valid link name value.

System programmer response
None.

Module
IUPKTRC

Procedure name
PARSEPKT

EZY0046E  IP=* invalid with SUBNET

Explanation
The SUBNET trace option cannot be specified with IP=*. An IP option of an asterisk (*) generates an implied subnet address mask (SUBNET option) of 0.0.0.0. In other words, IP=* is the equivalent of IP=0.0.0.0 SUBNET=0.0.0.0. The command is ignored. No changes are made to any trace options.

System action
Processing continues.

Operator response
Reenter the PKTTRACE command, and remove the SUBNET option, or specify an IP address other than *.
EZY0047E  Undefined linkname linkname

Explanation
The specified link name is not defined in the driver. The PKTTRACE LIST command can be used to display the defined link names for the driver. Note that SNALINK LU0 must have the IUCV connection open for the LINK before it is recognized. The command is ignored. No changes are made to any trace options.

System action
Processing continues.

Operator response
Reenter the PKTTRACE command, specifying a defined link name.

EZY0048E  Undefined keyword value option option

Explanation
The specified text is not a valid keyword value option for the PKTTRACE command. The command is ignored. No changes are made to any trace options.

System action
Processing continues.

Operator response
Reenter the PKTTRACE command, specifying a valid option. The valid keyword value options are:

- LINKNAME=
- PROT=
- IP=
- SUBNET=
- SRCPORT=
System programmer response
None.

Module
IUPKTTRC

Procedure name
PARSEPKT

EZY0052E PKTCLEAR: Trace entry list is already clear

Explanation
The PKTCLEAR subroutine has been called to clear the linked list of trace entries, however the list contained no entries. The PKTCLEAR subroutine is called when the device driver is terminating due to a shutdown request. This is an internal error. This error will not cause system problems.

System action
The driver continues to shut down.

Operator response
Contact the system programmer.

System programmer response
Contact the IBM Software Support Center to report this internal error.

Module
IUPKTTRC

Procedure name
PKTCLEAR

EZY0053E PKTCLEAR: FREEMAIN failed with return code return_code

Explanation
The PKTCLEAR subroutine has been called to clear the linked list of trace entries. The FREEMAIN system macro used to free the virtual storage for each entry has failed with the specified return code. The PKTCLEAR subroutine is called when the device driver is terminating due to a shutdown request. This might be an internal error. This error will not cause system problems.

System action
The driver continues to shut down.

Operator response
Contact the system programmer.
System programmer response
Use the return code from the FREEMAIN macro to identify the problem. See z/OS MVS Programming: Authorized Assembler Services Reference ALE-DYN for information about the FREEMAIN macro and its return codes. If the error still occurs, contact the IBM Software Support Center for more information.

Module
IUPKTTRC

Procedure name
PKTCLEAR

EZY0054E PKTLINK : GETMAIN failed with return code return_code

Explanation
The PKTLINK subroutine has been called to allocate an entry in the linked list of trace entries. The GETMAIN macro used to allocate the virtual storage for the entry has failed with the specified return code. The PKTLINK subroutine is called during device driver initialization. Packet tracing might be disabled for some or all links.

System action
Processing continues.

Operator response
Contact the system programmer.

System programmer response
Define more virtual storage and try to restart the driver. Use the return code from the GETMAIN macro to identify the problem. See z/OS MVS Programming: Authorized Assembler Services Reference ALE-DYN for information about the GETMAIN macro and its return codes. If the error still occurs, contact the IBM Software Support Center for more information.

Module
IUPKTTRC

Procedure name
PKTLINK

EZY0055E PKTLINK : Linkname linkname already in trace entry list

Explanation
The PKTLINK subroutine has been called to allocate an entry in the linked list of trace entries. The link name specified in the subroutine call is already present in the list of trace entries. The PKTLINK subroutine is called during device driver initialization. This is an internal error. This error will not cause system problems. Packet tracing might be disabled for some or all links.

System action
Processing continues.

Operator response
Contact the system programmer.
System programmer response
Contact the IBM Software Support Center to report this internal error.

Module
IUPKTTRC

Procedure name
PKTTRACE

**EZY0056E** PKTTRACE: No linknames defined

**Explanation**
The PKTTRACE subroutine has been called to determine if an IP packet should be traced, however the linked list of trace entries is empty. The list of trace entries should be initialized with an entry corresponding to each link that is defined for the device driver. This is an internal error. This error will not cause system problems. Packet tracing is disabled for all links.

**System action**
Processing continues.

**Operator response**
Contact the system programmer.

System programmer response
Contact the IBM Software Support Center to report this internal error.

Module
IUPKTTRC

Procedure name
PKTTRACE

**EZY0057E** PKTTRACE: Packet length does not include IP or UDP/TCP headers

**Explanation**
The PKTTRACE subroutine has been called to trace an IP packet that has been corrupted or truncated, removing headers that are required to determine if the packet is to be traced. The packet is not a valid IP packet. This could be a network problem or an internal error. This error will not cause system problems. The packet in error is not traced.

**System action**
Processing continues.

**Operator response**
Contact the system programmer.
**System programmer response**
The formatted output that appears prior to this message indicates which connection the IP packet belongs to. Use PKTTRACE to determine if a network node is creating an IP packet that is not valid or if part of the network is damaging IP packets.

**Module**
IUPKTTRC

**Procedure name**
PKTTRACE

**EZY0058E**  PKTTRACE : Linkname linkname not in trace entry list

**Explanation**
The PKTTRACE subroutine has been called to determine if an IP packet should be traced, however the link that the packet was sent to or received from is not in the linked list of trace entries. The list of trace entries should be initialized with an entry corresponding to each link that is defined for the device driver. This is an internal error. This error will not cause system problems. Packet tracing might be disabled for some or all links.

**System action**
Processing continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Contact the IBM Software Support Center to report this internal error.

**Module**
IUPKTTRC

**Procedure name**
PKTTRACE

**EZY0620E**  Incorrect function code specified for EZAPPFCF Routing: Major Key = key Minor Key = key

**Explanation**
The VSAM function request made to module EZAPPFCF is not supported. The program returns an error code to the calling module.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Obtain a dump and contact the IBM Software Support Center.
**Module**
EZAPPFCF

**Procedure name**
ERR00001

**EZY0621E**  
Storage Allocation Error in EZAPPFCF Routing: Major Key = `key` Minor Key = `key`

**Explanation**
Insufficient virtual storage available for EZAPPFCF processing.

**System action**
EZAPPFCF ends abnormally.

**Operator response**
None.

**System programmer response**
Increase the region size of the address space.

**Module**
EZAPPFCF

**Procedure name**
ERR00002

**EZY0622E**  
LOAD failure for EZAPPFCA Routing: Major Key = `key` Minor Key = `key`

**Explanation**
Module EZAPPFCA is unavailable.

**System action**
The program ends abnormally.

**Operator response**
None.

**System programmer response**
Make sure module EZAPPFCA is available through LINKLIST or STEPLIB specifications.

**Module**
EZAPPFCF

**Procedure name**
ERR00003

**EZY0630E**  
Specified Routing not in Routing File Routing: Major Key = `key` Minor Key = `key`
Explanation
The routing specified does not exist in the routing file. The program returns an error code to the calling module.

System action
Processing continues.

Operator response
If the routing was generated by JES, check the class, destination, and forms specified on the JES DD statement. Unless modified by an exit, a JES routing takes the form DDDDD0000CFFFF000. The characters represent class, destination and forms as indicated in the following:

C
Class

DDDD
Destination

FFFF
Forms

If the routing was generated by VTAM, check the LU name and LU type. Unless modified by an exit, a VTAM routing takes the form LUNAMExxxxxxxx. The LU name is represented in this form as follows:

LUNAMExx
The LU name.

xxxxxxxx
Any string, but it must be specified.

If the exit is a general routing exit, specific routing exit, or input record exit, modify the routing name.

System programmer response
If the exit was a general routing exit, specific routing exit, or input record exit, modify the routing name and check the logic of the exit for errors.

Module
EZAPPFCF

Procedure name
ERR01001

EZY0631E Error in General Routing Exit RC=xxxx. Routing: Major Key = key Minor Key = key

Explanation
The general routing exit returned an error condition indicator in register 15. Any return codes other than zero, one, or two indicate the exit did not complete successfully.

System action
Processing continues.

Operator response
None.
System programmer response
Determine the cause of the error condition indicated by the EZAPPGPR return code. If necessary, restart the capture point after correcting the error.

Module
EZAPPFCF

Procedure name
ERR01002

EZY0632E  Error in Specific Routing Exit =xxxx. Routing: Major Key = key Minor Key = key

Explanation
The specific routing exit returned with a nonzero value xxxx in register 15.

System action
Processing continues.

Operator response
None.

System programmer response
Use the error code to determine the cause of the problem and correct it.

Module
EZAPPFCF

Procedure name
ERR01003

EZY0633E  Number of routing records not equal to number of destinations Routing: Major Key = key Minor Key = key

Explanation
The number of routing records found for this key is not equal to the number of records specified in the type N record. The program returns an error code to the calling module.

System action
Processing continues.

Operator response
Correct the routing file.

System programmer response
None.

Module
EZAPPFCF
**Procedure name**

ERR01004

**EZY0634E Specified Options not in Options File Routing: Major Key = key Minor Key = key**

**Explanation**
The option specified does not exist in the options file. The program returns an error code to the calling module.

**System action**
Processing continues.

**Operator response**
Check the routing file entry for a correctly specified options entry.

**System programmer response**
If this is a general routing exit, specific routing exit, or input record exit, modify the routing name, and check the logic of the exit for errors.

**Module**
EZAPPFCF

**Procedure name**

ERR01005

**EZY0635E VSAM Error on operation Return Code=rc Reason Code=reason DDNAME=ddname**

**Explanation**
An error was encountered in a VSAM operation. The values specified in the message indicate the following:

- **operation**
  The operation that proved unsuccessful

- **rc**
  The return code

- **reason**
  The reason code

- **ddname**
  The ddname of the file

The program returns an error code to the calling module.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
See z/OS DFSMS Macro Instructions for Data Sets for an explanation of the reason and response codes.
**Module**
EZAPPFCF

**Procedure name**
VSMERR50

**EZY0636E  LINK Error on Specific Routing Exit.**

**Explanation**
The specific routing exit defined in this routing cannot be accessed. Routing: Major Key = *key* Minor Key = *key*. The program returns to the caller with an error code.

**System action**
Processing continues.

**Operator response**
Do the following:
- Note message CSV003I which gives the name of the module not found.
- Make sure the exit is correctly named in the routing file definition.
- Make sure the exit module exists in STEPLIB or JOBLIB or LPA data sets.

**System programmer response**
None.

**Module**
EZAPPFCF

**Procedure name**
ERR01006

**EZY0651E  Invalid function code passed to EZAPPFCM Routing: Major Key = *key* Minor Key = *key***

**Explanation**
EZAPPFCM was invoked with a function code that is not valid. EZAPPFCM returns a X'0001' return code to the caller.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
If the user is invoking EZAPPFCM directly, check the invoking program for a valid function code. If EZAPPFCM is being invoked by either the NPF FSS writer or NPF VTAM capture point application, contact the IBM Software Support Center.
Module
EZAPPFCM

Procedure name
ERR00001

**EZY0652E**  Storage allocation error in EZAPPFCM Routing: Major Key = *key* Minor Key = *key*

**Explanation**
An attempted GETMAIN in EZAPPFCM was unsuccessful. EZAPPFCM returns a X'0002' return code to the caller.

**System action**
Processing continues.

**Operator response**
Increase the virtual storage allocation for the address space and resubmit the job.

**System programmer response**
None.

Module
EZAPPFCM

Procedure name
ERR00002

**EZY0653E**  Unable to load EZAPPFCF Routing: Major Key = *key* Minor Key = *key*

**Explanation**
The attempt to load module EZAPPFCF was unsuccessful. EZAPPFCM returns a X'0003' return code to the caller.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Check the libraries in the STEPLIB or JOBLIB DD statement to make sure one of them contains the EZAPPFCF load module.

Module
EZAPPFCM

Procedure name
ERR00003

**EZY0654E**  Error loading input record exit Routing: Major Key = *key* Minor Key = *key*
**Explanation**
The options data specified an input record exit, but the attempt to load it was unsuccessful. EZAPPFCM returns a X'0004' return code to the caller.

**System action**
Processing continues.

**Operator response**
1. Determine if the routing is correct.
2. Check the routing file entry for this routing. The options name specified must match an options file record.

**System programmer response**
1. Check the JOBLIB and STEPLIB concatenations to make sure the exit module is available.
2. If any user exits modify the routing data area, check the logic of the user exits.

**Module**
EZAPPFCM

**Procedure name**
ERR00004

---

**EZY0655E**   Unable to attach EZAPPLPR Routing: Major Key = key Minor Key = key

**Explanation**
The attempt by module EZAPPFCM to ATTACH the EZAPPLPR subtask was unsuccessful. EZAPPFCM returns a X'0005' return code to the caller.

**System action**
For the VTAM capture point, processing continues. For the JES capture point, the program ends abnormally.

**Operator response**
Notify the system programmer.

**System programmer response**
1. Check the JOBLIB and STEPLIB concatenations to make sure that module EZAPPLPR is available.
2. Contact the IBM Software Support Center.

**Module**
EZAPPFCM

---

**EZY0656E**   Error on Dynamic Allocation of data file, RC=return code, S99RET=code DSname =name

**Explanation**
Attempts to allocate a print data file were unsuccessful. The dynamic allocation return code and reason code are displayed. If the failure occurred during SMS processing, the SMS reason code (S99ERSN) displays in place of the DYNALLOC reason code (S99ERROR).
**System action**
The dynamic allocation will be tried again. If that attempt is also unsuccessful, an error condition will be returned to the capture point indicating a print data set is not available.

**Operator response**
Check for related messages that indicate insufficient DASD space or insufficient virtual storage and correct the problem indicated.

**System programmer response**
Investigate the reason for any dynamic allocation return codes contained in the message. These return and reason codes are documented in z/OS MVS Programming: Authorized Assembler Services Guide.

**Module**
EZAPPFCM

**Procedure name**
ERR01001

**EZY0657E** OPEN failure on data file. DSname =name

**Explanation**
The print data file named could not be opened. EZAPPFCM returns a X'1003' return code to the caller.

**System action**
Processing continues.

**Operator response**
Look for associated messages and correct the problems indicated.

**System programmer response**
None.

**Module**
EZAPPFCM

**Procedure name**
ERR01003

**EZY0658E** CLOSE failure on data file. DSname =name

**Explanation**
The print data file named could not be closed. EZAPPFCM returns a X'2001' return code to the caller.

**System action**
Processing continues.

**Operator response**
Look for associated messages and correct the problems indicated.
System programmer response
None.

Module
EZAPPFCM

Procedure name
ERR02001

**EZY0659E** PUT failure on data file. DSname =name

**Explanation**
An error occurred on a PUT to the named file. EZAPPFCM returns a X'3002' return code to the caller.

**System action**
Processing continues.

**Operator response**
Look for associated messages and correct the problems indicated.

System programmer response
None.

Module
EZAPPFCM

Procedure name
ERR03002

**EZY0660E** Error on input record exit Routing: Major Key = key Minor Key = key

**Explanation**
A user-provided input record exit returned a nonzero return code in Register 15. EZAPPFCM returns a X'3003' return code to the caller.

**System action**
Processing continues.

**Operator response**
None.

System programmer response
Determine the problem with the input record exit and correct it.

Module
EZAPPFCM
**Procedure name**
ERR03003

**EZY0661E** Record length error Routing: Major Key = key Minor Key = key

**Explanation**
A record length less than 0 was specified. EZAPPFCM returns a X'3004' return code to the caller.

**System action**
Processing continues.

**Operator response**
Determine the source of the incorrect record length and correct it.

**System programmer response**
None.

**Module**
EZAPPFCM

---

**Procedure name**
ERR03004

**EZY0662E** Data file open for SEND DSname = name

**Explanation**
EZAPPFCM received a request to send the data set to LPR but it is still open. Data might be lost. EZAPPFCM returns a X'4001' return code to the caller.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
If this is a user-written invocation of EZAPPFCM, make sure the CLOSE function was invoked prior to doing the SEND. If EZAPPFCM is invoked by the NPF FSS writer or NPF VTAM capture point, notify the IBM Software Support Center.

**Module**
EZAPPFCM

---

**Procedure name**
ERR04001

**EZY0663E** Data file deallocation error DSname = name
Explanation
An error occurred when deallocating the named data set. EZAPPFCM returns a X'4004' return code to the caller.

System action
Processing continues.

Operator response
None.

System programmer response
Notify the IBM Software Support Center.

Module
EZAPPFCM

Procedure name
ERR04004

EZY0664E  Error on Queue record creation DSname =name

Explanation
An error occurred on the creation of the queue record associated with the named data set. EZAPPFCM returns a X'4004' return code to the caller.

System action
Processing continues.

Operator response
Look for associated message EZY0635E to determine the cause of the problem.

System programmer response
None.

Module
EZAPPFCM

Procedure name
ERR04004

EZY0665E  Capture Point Initialization Failure

Explanation
A failure occurred at startup of the VTAM or JES capture points or during TCPIP initialization while processing in the capture points. The error is returned to the caller.

System action
The program ends abnormally.
Operator response
See the accompanying message to determine the cause of the failure.

System programmer response
None.

Module
EZAPPFCM

EZY0666I File Management Initialization Completed

Explanation
Network Print Facility file management initialization is complete. All parameters specified were acceptable.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFCM

EZY0667I PRINT STARTUP USED DATASET name

Explanation
The startup process for Network Print Facility used the indicated data set as a source for one or more of the startup parameters.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFCI

EZY0668E INVALID HIGH LEVEL QUALIFIER

Explanation
The high-level qualifier for the data set name is not valid. The high-level qualifier contains one of these errors:
• The qualifier is longer than 26 characters.
• The qualifier starts or ends with a period.
• The qualifier has a node name longer than 8 characters.

System action
Processing continues.

Operator response
Enter a valid high-level qualifier for the data set name and reissue the request.

System programmer response
None.

Module
EZAPPFCI

EZY0669E  INVALID TCP/IP JOBNAME

Explanation
The job name specified is not valid; it is longer than 8 characters.

System action
Processing continues.

Operator response
Enter a valid job name and reissue the request.

System programmer response
None.

Module
EZAPPFCI

EZY0670E  INVALID ALLOCATION PARAMETER

Explanation
The allocation parameter specified is not valid. The allocation parameter contains one of these errors:
• The type is not CYL, TRK, or a number.
• The number specified for a block size is zero or negative.
• The primary allocation is not positive or not specified.
• The secondary allocation is negative or not specified.

System action
Processing continues.

Operator response
Make sure the allocation parameters are entered correctly and reissue the request.
System programmer response
None.

Module
EZAPPFCI

**EZY0671E** INVALID THREAD SPECIFICATION, n WAS USED

**Explanation**
The specified thread value was not valid. The thread value must be greater than, or equal to one and less than, or equal to eleven. The value in n will be used instead.

**System action**
Processing continues.

**Operator response**
Specify a valid thread value if the default value in n is not acceptable.

System programmer response
None.

Module
EZAPPFCI

**EZY0672E** DEFAULT HIGH LEVEL QUALIFIER INVALID

**Explanation**
The specified high-level qualifier in module CMMVSYS is longer than 26 characters.

**System action**
Processing continues.

**Operator response**
Make sure the high-level qualifier in CMMVSYS is not longer than 26 characters and reissue the request.

System programmer response
None.

Module
EZAPPFCI

**EZY0673E** Insufficient space for data file.

**Explanation**
Network Print Facility did not have enough space to create the data file. This results in Network Print Facility issuing an x37 abend and a 4005 return code.

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

System programmer response
Increase the space allocation parameter values used for startup.

Module
EZAPPFCM

EZY0674E  PUT attempted on file which is not open

Explanation
The Network Print Facility has found an error attempting to write to the QSAM print file. A return code of x'3006' is passed back to the caller.

System action
Processing continues.

Operator response
None.

System programmer response
Look for associated messages and investigate the reason why the print file might not have been opened or allocated.

Module
EZAPPFCM

EZY0676I  Startup value for dataset value

Explanation
This message displays the value passed to the JES writer on startup for data sets created by the JES writer.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFCI

Procedure name
main.

EZY0677E  Error on deletion of Active record
Explanation
An error was encountered while attempting to delete a record from the active file.

System action
Processing continues.

Operator response
This error will be accompanied by message EZY0635E. See that message for diagnosis information.

System programmer response
None.

Module
EZAPPFCM

EZY0678E  NON-ZERO RETURN CODE xxxxxxxx FROM INITAPI CALL TO yyyyyyyy

Explanation
A failure occurred at startup of the VTAM or JES capture points, or during Queue Manager initialization while attempting to initialize the SOKETS interface to the TCPIP address space. The error code from the INITAPI call (xxxxxxxx) and the TCPIP address space name used for the call (yyyyyyyy) are displayed. The error is returned to the caller.

System action
The program ends abnormally.

Operator response
Ensure that the TCPIP address space is active prior to starting the VTAM or JES capture points or the Queue Manager. An error of 00010191 indicates that the TCPIP IUCV was not installed.

System programmer response
Ensure that the TCPIPJOBNAME value matches the job name of the currently active TCPIP address space. If no TCPIPJOBNAME statement is found in any of the parameter data sets, the default value of TCPIP is used. For information on the extended socket return codes xxxxxxxx, see z/OS Communications Server: IP and SNA Codes.

Module
EZAPPFCI

EZY0679I  parameter Default value of yyyyyyyyy used

Explanation
This message displays the default value yyyyyyyyy used by the capture points or Queue manager on startup for the TCPIP global parameters. The parameter field will be either TCPIPJOBNAME or DATASETPREFIX.

System action
Processing continues.

Operator response
None.
System programmer response
None.

Module
EZAPPFCI

EZY0680I  Network Print Facility Recovery Started

Explanation
The active file contained at least one meaningful record. Each active file record that points to a closed temporary print data set will be copied into a queue file record and then deleted. All other active file records will be deleted, along with their corresponding temporary print data sets.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFCM

EZY0681I  Network Print Facility Recovery Completed

Explanation
All records in the active file were processed.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFCM

EZY0682E  INVALID UNIT SPECIFICATION

Explanation
The UNIT name specified on the NPFUNIT statement is out of range. The range of valid UNIT values is from one to eight characters.
System action
The error is returned to the caller.

Operator response
Correct the NPFUNIT name statement in NPF.DATA. For further information about the UNIT parameter, see the z/OS MVS JCL Reference.

System programmer response
Ensure that the name specified on the NPFUNIT statement is a valid esoteric unit name, generic unit name, or unit address defined in the EDT for this system.

Module
EZAPPFCI

EZY0700E INSUFFICIENT DESTINATIONS FOR PREVIOUS ROUTING

Explanation
The previous EZAPPFL TYPE=ROUTING macro specified more than 1 destination but there are not enough EZAPPFL TYPE=NXTDEST macros to describe them.

System action
Macro generation stops.

Operator response
Either correct the number of destinations specified in the EZAPPFL TYPE=ROUTING macro or add EZAPPFL TYPE=NXTDEST macros.

System programmer response
None.

Module
EZAPPFL

Procedure name
.TYPE000

EZY0701E EXCESSIVE DESTINATIONS FOR PREVIOUS ROUTING

Explanation
An EZAPPFL TYPE=NXTDEST macro was encountered when it was not expected.

System action
Macro generation stops.

Operator response
Check the NDEST parameter of the previous EZAPPFL TYPE=ROUTING macro. Make sure that the number of destinations is equal to the number of EZAPPFL TYPE=NXTDEST macros following the EZAPPFL TYPE=ROUTING macro plus 1.
System programmer response
None.

Module
EZAPPFL

Procedure name
.TYPE000

Ezy0702s  NO TYPE SPECIFIED IN EZAPPFL MACRO

Explanation
The EZAPPFL macro did not contain a recognizable TYPE parameter.

System action
Macro generation stops.

Operator response
Correct and reissue the macro.

System programmer response
None.

Module
EZAPPFL

Procedure name
.TYPE010

Ezy0703e  NO MAJOR KEY SPECIFIED FOR ROUTING

Explanation
The EZAPPFL TYPE=ROUTING macro did not contain a MAJKEY parameter. This parameter is mandatory.

System action
Macro generation stops.

Operator response
Correct and reissue the macro.

System programmer response
None.

Module
EZAPPFL
EZY0704E  LENGTH OF MAJOR KEY GREATER THAN 8

Explanation
The MAJKEY parameter of the EZAPPFL TYPE=ROUTING macro contained more than 8 characters. The maximum length for the MAJKEY is 8.

System action
Macro generation stops.

Operator response
Correct and reissue the macro.

System programmer response
None.

Module
EZAPPFL

Procedure name
.ROUT005

EZY0705E  NO MINOR KEY SPECIFIED FOR ROUTING

Explanation
The EZAPPFL TYPE=ROUTING macro did not contain a MINKEY parameter. This parameter is mandatory.

System action
Macro generation stops.

Operator response
Correct and reissue the macro.

System programmer response
None.

Module
EZAPPFL

Procedure name
.ROUT010

EZY0706E  LENGTH OF MINOR KEY GREATER THAN 8

Explanation
The MINKEY parameter of the EZAPPFL TYPE=ROUTING macro contained more than 8 characters. The maximum length for MINKEY is 8.
System action
Macro generation stops.

Operator response
Correct and reissue the macro.

System programmer response
None.

Module
EZAPPFL

Procedure name
.ROUT015

EZY0707E ROUTING KEY SEQUENCE ERROR
PREVIOUS: MAJKEY=xxxxxxxx MINKEY=yyyyyyyy
CURRENT: MAJKEY=wwwwwwww MINKEY=zzzzzzzz

Explanation
The key specified in the current EZAPPFL TYPE=ROUTING is lower in collating sequence than the previous key. The routing macros must be in ascending order based on their MAJKEY and MINKEY parameter values.

System action
Macro generation stops.

Operator response
Rearrange and reissue the routing macros.

System programmer response
None.

Module
EZAPPFL

Procedure name
.ROUT020

EZY0708I NUMBER OF DESTINATIONS NOT SPECIFIED, ASSUMED AS 1.

Explanation
The NDEST parameter was not specified in the EZAPPFL TYPE=ROUTING macro. The macro assumes a value of 1.

System action
Macro generation continues.

Operator response
None, if 1 is the correct value. Otherwise, specify the NDEST parameter and reissue the macro.
**System programmer response**
None.

**Module**
EZAPPFL

**Procedure name**
.ROUT030

**EZY0709E INVALID VALUE IN VTCLASS**

**Explanation**
The VTCLASS parameter in the EZAPPFL TYPE=ROUTING macro specified a class less than 1 or greater than 64.

**System action**
Macro generation stops.

**Operator response**
Correct the class and reissue the macro.

**System programmer response**
None.

**Module**
EZAPPFL

**Procedure name**
.ROUT060

**EZY0710E OPTNAME PARAMETER REQUIRED FOR ROUTING**

**Explanation**
The OPTNAME parameter was not specified in the EZAPPFL TYPE=ROUTING macro. This parameter is required.

**System action**
Macro generation stops.

**Operator response**
Correct and reissue the macro.

**System programmer response**
None.

**Module**
EZAPPFL
**EZY0711E  SREXIT NAME TOO LONG**

**Explanation**
The SREXIT (specific routing exit) parameter in the EZAPPFL TYPE=ROUTING macro specified a name longer than 8 characters.

**System action**
Macro generation stops.

**Operator response**
Correct the SREXIT parameter and reissue the EZAPPFL TYPE=ROUTING macro.

**System programmer response**
None.

**Module**
EZAPPFL

**Procedure name**
.ROUT110

**EZY0712E  INAME PARAMETER REQUIRED FOR ROUTING.**

**Explanation**
The INAME (Internet name/address) parameter was not specified in the EZAPPFL TYPE=ROUTING macro. This parameter is required.

**System action**
Macro generation stops.

**Operator response**
Correct and reissue the macro.

**System programmer response**
None.

**Module**
EZAPPFL

**Procedure name**
.ROUT125

**EZY0713E  PNAME PARAMETER REQUIRED FOR ROUTING.**
Explanation
The PNAME (printer name) parameter was not specified in the EZAPPFL TYPE=ROUTING macro. This parameter is required.

System action
Macro generation stops.

Operator response
Correct and reissue the macro.

System programmer response
None.

Module
EZAPPFL

Procedure name
.ROUT125

Explanation
The INAME (Internet name/address) parameter was not specified in the EZAPPFL TYPE=NXTDEST macro. This parameter is required.

System action
Macro generation stops.

Operator response
Correct and reissue the macro.

System programmer response
None.

Module
EZAPPFL

Procedure name
.NEXT010

Explanation
The PNAME (printer name) parameter was not specified in the EZAPPFL TYPE=NXTDEST macro. This parameter is required.

System action
Macro generation stops.
### EZY0716E  OPTNAME PARAMETER REQUIRED FOR OPTIONS

**Explanation**
The OPTNAME parameter was not specified in the EZAPPFL TYPE=OPTIONS macro. This parameter is required.

**System action**
Macro generation stops.

**Operator response**
Correct and reissue the macro.

**System programmer response**
None.

---

### EZY0717E  IREXIT NAME TOO LONG

**Explanation**
The IREXIT (input record exit name) parameter specified in the EZAPPFL TYPE=OPTIONS was longer than 8 characters.

**System action**
Macro generation stops.

**Operator response**
Correct the name and reissue the EZAPPFL TYPE=OPTIONS macro.

**System programmer response**
None.
**Explanation**

The Network Print Facility file load program has determined that the key specified in the current EZAPPFL TYPE=OPTIONS macro is lower in collating sequence than the key for the previous option name. The options macros must be specified in ascending order based on their OPTNAME parameter value.

**System action**

Macro generation stops.

**Operator response**

Rearrange the EZAPPFL TYPE=OPTIONS macros in ascending order by OPTNAME and resubmit the job.

**System programmer response**

None.

---

**Explanation**

The Network Print Facility or one of the CICS® Sockets file load programs has received a nonzero return code from VSAM services. The VSAM return and reason codes are displayed in the message following the failing VSAM macro (macro=OPEN, PUT, or CLOSE) and file type (file=ROUTING, OPTIONS, or QUEUE).

For VSAM PUTs, the error message will also include the key values, either majkey and minkey for the routine file or optname for the options file.

**System action**

Macro generation stops.

**Operator response**

Correct the error and resubmit the job.

**System programmer response**

Investigate the reason for the nonzero VSAM return codes displayed in the error message. These return and reason codes are documented in z/OS DFSMS Macro Instructions for Data Sets.
**Module**
EZAPPFL, EZACICD, EZACICR

**Procedure name**
n/a

**EZY0720S** OPTIONS ENTRY PRECEDES ROUTING ENTRY

**Explanation**
AN EZAPPFL TYPE=OPTIONS macro was encountered prior to an EZAPPFL TYPE=ROUTING macro. All routing entries must precede the first options entry.

**System action**
Macro generation stops.

**Operator response**
Correct the sequence of the macros and reissue them.

**System programmer response**
None.

**Module**
EZAPPFL

**Procedure name**
.ROUT001

**EZY0721W** NDEST EXCEEDS 32767

**Explanation**
The number of destinations specified exceeds the maximum value of 32767. The value is replaced with 32767.

**System action**
Processing continues.

**Operator response**
Correct the NDEST and resubmit the job.

**System programmer response**
Assist the user in submitting an accurate NDEST parameter value. For more information on the NDEST parameter see z/OS Communications Server: IP Network Print Facility.

**Module**
EZAPPFL

**Procedure name**
MNOTE

**EZY0722E** OPTNAME PARAMETER EXCEEDS 16 CHARACTERS
Explanation
The specified OPTNAME parameter exceeds 16 characters in length. Macro generation is terminated.

System action
Processing continues.

Operator response
Correct the OPTNAME and resubmit the job.

System programmer response
Assist the user in submitting an accurate OPTNAME parameter value. For more information on the OPTNAME parameter see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPFL

Procedure name
MNOTE

EZY0723E RETRY LIMIT EXCEEDS 32767

Explanation
The specified retry limit parameter exceeds the maximum value of 32767. The value is replaced with 32767.

System action
Processing continues.

Operator response
Correct the RETRY LIMIT parameter and resubmit the job.

System programmer response
Assist the user in submitting an accurate RETRY LIMIT parameter value. For more information on the RETRY LIMIT parameter see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPFL

Procedure name
MNOTE

EZY0730E TIME VALUE TOO LONG

Explanation
The value specified for either RETAINT or RETRYT exceeds 8 digits. The system sets the time value to zero.

System action
Processing continues.
Operator response
Correct the time value and reissue RETAINT or RETRYT.

System programmer response
None.

Module
EZAPPFLT

Procedure name
.L9

**EZY0731E**  INVALID CHARACTER IN TIME VALUE

Explanation
The EZAPPFLT macro detected a nonnumeric character in the time value. The system sets the time value to zero.

System action
Processing continues.

Operator response
Correct the time value and reissue the macro.

System programmer response
None.

Module
EZAPPFLT

Procedure name
.NERR1

**EZY0732E**  INVALID DAY VALUE

Explanation
The EZAPPFLT macro detected a day value greater than 366. The system sets the time value to zero.

System action
Processing continues.

Operator response
Correct the day value and reissue the macro.

System programmer response
None.
Module
EZAPPFLT

Procedure name
.N2

**EZY0733E** INVALID HOUR VALUE

**Explanation**
The EZAPPFLT macro detected an hour value greater than 23. The system sets the time value to zero.

**System action**
Processing continues.

**Operator response**
Correct the hour value and reissue the macro.

**System programmer response**
None.

Module
EZAPPFLT

Procedure name
.NERR3

**EZY0734E** INVALID MINUTE VALUE

**Explanation**
The macro detected a minute value greater than 59. The system sets the time value to zero.

**System action**
Processing continues.

**Operator response**
Correct the minute value and reissue the macro.

**System programmer response**
None.

Module
EZAPPFLT

Procedure name
.NERR4

**EZY0750I** NPF Logging Facility Started
**Explanation**
The Network Print Facility has started the logging activity, which will write all output to the data set specified by the EZAPPLOG DD statement.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFCM

**EZY0751E OPEN Failure on NPF Logging File**

**Explanation**
An error was encountered while attempting to open the data set specified by the EZAPPLOG DD statement. No logging will be done for this instance of NPF.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Verify that the file specified by the EZAPPLOG DD statement exists and is correctly specified.

**Module**
EZAPPFCM

**EZY0752E No More Space for NPF Logging File**

**Explanation**
An error was encountered while attempting to write a record to the data set specified by the EZAPPLOG DD statement. All logging will be terminated immediately for this instance of NPF.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Increase the size of the data set specified by the EZAPPLOG DD statement and restart NPF.
Module
EZAPPFCM

**EZY0753E I/O Error on NPF Logging File**

**Explanation**
The Network Print Facility has encountered an I/O error while attempting to write a record to the data set specified by the EZAPPLOG DD statement. All logging will be ended immediately for this instance of NPF.

**System action**
NPF processing continues.

**Operator response**
None.

**System programmer response**
Investigate the cause of the I/O error associated with the EZAPPCTR DD statement. Previous error messages might have been generated.

Module
EZAPPCTR

**Procedure name**

**EZY0800I Starting the Network Print Facility VTAM Interface (NPFV)**

**Explanation**
The Network Print Facility VTAM interface has been invoked.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

Module
EZAPPAAA

**Procedure name**
AACMSG00

**EZY0801I Terminating the Network Print Facility VTAM Interface (NPFV)**

**Explanation**
The Network Print Facility VTAM interface is terminating.
**System action**
The program ends.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPAAA

**Procedure name**
AAAEXIT0

---

**EZY0803E**  Storage allocation error

**Explanation**
The NPF VTAM capture point application encountered a storage allocation error.

**System action**
The program ends abnormally.

**Operator response**
Notify the system programmer.

**System programmer response**
Increase the storage allocation for the capture point.

**Module**
EZAPPAAA, EZAPPCHD, EZAPPCLU, EZAPPOPN, EZAPPQRS, EZAPPQST, EZAPPVPP

**Procedure name**
various

---

**EZY0804E**  Timer not operational

**Explanation**
A timer used by the NPF VTAM capture point application is not operational. This can indicate a programming error.

**System action**
The application continues.

**Operator response**
None.
**System programmer response**
If this error results in system problems, contact the IBM Software Support Center.

**Module**
EZAPPCHD

**Procedure name**
*various*

**EZY0806E No PARM field specified**

**Explanation**
The EXEC statement for the NPF VTAM capture point application did not contain the required PARM value field.

**System action**
The program ends abnormally.

**Operator response**
None.

**System programmer response**
Correct and reissue the EXEC statement. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

**Module**
EZAPPPRS

**Procedure name**
NOPARMS, PARMSOK

**EZY0807E PARM list ended abnormally**

**Explanation**
MVS was unable to create a PARM list for use by the NPF VTAM capture point application. This indicates an internal error.

**System action**
The application halts.

**Operator response**
Notify the system programmer.

**System programmer response**
Contact the IBM Software Support Center.

**Module**
*various*
**EZY0808E**  No keyword specified in the PARM field

**Explanation**
The EXEC statement for the NPF VTAM capture point application did not contain a required keyword before the equal sign in the PARM list.

**System action**
The program ends abnormally.

**Operator response**
None.

**System programmer response**
Correct the PARM list and reissue the EXEC statement. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

**Module**
EZAPPFRS

**Procedure name**
GOTKEY

**EZY0809E**  Keyword too long: keyword

**Explanation**
The EXEC statement for the NPF VTAM capture point application contained a keyword in the PARM field that was greater than 8 characters.

**System action**
The program ends abnormally.

**Operator response**
None.

**System programmer response**
Correct the PARM field and reissue the EXEC statement. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

**Module**
EZAPPFRS

**Procedure name**
GOODKEYL

**EZY0810E**  Invalid keyword: keyword
Explanation
The EXEC statement for the NPF VTAM capture point application contained a keyword in the PARM field that was not valid.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Correct the PARM field and reissue the EXEC statement. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPPRS

Procedure name
FINDVALU

EZY0811E  No parameter specified for keyword

Explanation
The EXEC statement for the NPF VTAM capture point application contained a keyword in the PARM list value with no parameter specified.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Correct the PARM field completely and reissue the EXEC statement. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPPRS

Procedure name
PROCVALU

EZY0812E  Invalid parameter specified for keyword

Explanation
The EXEC statement for the NPF VTAM capture point application contained a keyword with an incorrect parameter.
System action
The program ends abnormally.

Operator response
None.

System programmer response
Correct the PARM field and reissue the EXEC statement. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPPRS

Procedure name
TRCLENOK

EZY0813E  parameter parameter too long

Explanation
The EXEC statement for the NPF VTAM capture point application contained a keyword with that was too long.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Correct the PARM field and reissue the EXEC statement. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPPRS

Procedure name
TRCLENOK

EZY0814E  No LUCLASS specified

Explanation
The PARM field on the EXEC statement did not specify an LUCLASS parameter. For the NPF VTAM capture point application, this parameter is required.

System action
The program ends abnormally.

Operator response
None.
System programmer response
Correct the PARM field and reissue the EXEC statement. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPPRS

Procedure name
PRSEXIT0

**EZY0815E**  No LUs were found with the specified LUCLASS

Explanation
No match was found between the LU classes specified in the startup parameters for a VTAM application and the LU classes specified for destinations in the routing file.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Correct the startup parameter or the routing file. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPCLU

Procedure name
CLUEXIT0

**EZY0816E**  MAXOPEN too big for MAXFLSTG

Explanation
The MAXOPEN and MAXFLSTG subparameters were both specified in the JCL that starts the Network Print Facility's VTAM Capture Point application. MAXFLSTG is less than MAXOPEN multiplied by 16K (MAXOPEN * 16K) which means that MAXOPEN is not providing any meaningful limit function.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Specify a MAXFLSTG value greater than or equal to MAXOPEN * 16K or eliminate the MAXOPEN subparameter. For more information on the MAXOPEN and MAXFLSTG subparameters, see z/OS Communications Server: IP Network Print Facility.
Module
EZAPPCHD

EZY0817E Unable to load module

Explanation
There was a problem loading the specified module.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Make sure that there is sufficient storage available to load the module.

Module
various

EZY0818E ATTACH of module failed

Explanation
The NPF VTAM capture point application was unable to attach the specified module.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Make sure that there is sufficient storage available to load the module.

Module
EZAPPVPP

Procedure name
VPCMSG07, ATTCHQST, ATTCHQOK

EZY0820E OPEN failed for routing file.

Explanation
The NPF VTAM capture point application was unable to open the routing file.

System action
The program ends abnormally.
**Operator response**
None.

**System programmer response**
Make sure the file indicated in the message exists and the DD statement is correct.

**Module**
EZAPPCLU

**Procedure name**
ISSUERR

**EZY0821E**  GET failed for routing file

**Explanation**
The NPF VTAM capture point application encountered an error trying to retrieve a routing record.

**System action**
If the program was initializing when the error occurred, the program ends abnormally, otherwise, the program continues.

**Operator response**
Notify the system programmer.

**System programmer response**
This should be accompanied by message EZY0630E, which gives the name of the routing that proved unsuccessful. Follow the response for that message.

**Module**
EZAPPCLU

**Procedure name**
ISSUERR

**EZY0822E**  name Duplicate LU names in routing file

**Explanation**
The Network Print Facility VTAM capture point has found multiple VTAM routings with the same major name (name). There cannot be duplicate names in the routing file.

A VTAM routing is one for which a nonzero LUCLASS value has been specified. For VTAM routings, the major name is the printer’s LU name. For more information on input record fields of the routing file, see the z/OS Communications Server: IP Network Print Facility manual.

**System action**
The program uses the first VTAM routine with major name name and ignores the duplicates.

**Operator response**
None.
System programmer response
Remove the duplicate name from the major name field of the routing file.

Module
EZAPPCLU

Procedure name
BADLU

EZY0823E  xxxxxxxx - Undetermined OPEN error, RC = rc

Explanation
An attempt to open an LU failed for an unknown reason. The value returned in the error field of the ACB is displayed in the rc portion of this message.

System action
The program marks the LU as permanently dead.

Operator response
Determine what type of error occurred by looking in the z/OS Communications Server: SNA Programming under the OPEN macro, Completion Information section, then contact the system programmer.

System programmer response
For error code 36 (24 hexadecimal), correct the specified APPL definition. For all other error codes, contact the IBM Software Support Center.

Module
EZAPPCLU

Procedure name

EZY0824E  name - LU unknown to VTAM

Explanation
The indicated LU is not known to VTAM.

System action
The program marks the LU as retryable and will periodically attempt to open the LU.

Operator response
Make sure the specified LU is defined to VTAM and activated.

System programmer response
Make sure the LUCLASS parameter in the JCL specifies the correct set of LU names in the routing file. Correct the startup parameter or the routing file. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.
**Module**
EZAPPOPN

**Procedure name**
OPCMSG05

**EZY0825E**  
**name** - LU is in the process of being opened or closed

**Explanation**
A connection request has been received for the specified LU that is in the process of opening or closing.

**System action**
The program marks the LU as retryable and will periodically attempt to open the LU.

**Operator response**
Make sure the LU is not already in use by another application.

**System programmer response**
None.

**Module**
EZAPPOPN

**Procedure name**
OPCMSG06

**EZY0826E**  
**LU name** - LU already in use by another application

**Explanation**
A connection request has been received for the specified LU which is already active.

**System action**
The system marks the LU as retryable and will periodically attempt to open the LU.

**Operator response**
None.

**System programmer response**
Correct the startup parameter or the routing file. Also, make sure the specified LU is not already in use by another application. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

**Module**
EZAPPOPN

**Procedure name**
OPCMSG07

**EZY0827E**  
**name** - LU was improperly defined in VTAM definition deck

---

70  z/OS Communications Server: IP Messages: Volume 3 (EZY)
**Explanation**  
The specified LU name that is assigned to the NPF VTAM capture point application has attributes that are inconsistent with this use.

**System action**  
The program ignores the LU.

**Operator response**  
Make sure the LU is not already in use.

**System programmer response**  
Correct the startup parameter or the routing file. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

---

**Module**  
EZAPPOPN

**Procedure name**  
OPCMMSG08

**EZY0828E**  
*number* LUs unknown to VTAM

**Explanation**  
The indicated number of LUs are not known to VTAM.

**System action**  
The program marks the LUs as retryable and will periodically attempt to open them.

**Operator response**  
Make sure the LUs are defined to VTAM and activated.

**System programmer response**  
Make sure the LUCLASS parameter in the JCL specifies the correct set of LU names in the routing file. Correct the startup parameter or the routing file. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

---

**Module**  
EZAPPOPN

**Procedure name**  
OPCMMSG05

**EZY0829E**  
*number* LUs in the process of being opened or closed

**Explanation**  
The indicated number of LUs are in the process of opening or closing.
System action
The program marks the LUs as retryable and will periodically attempt to open them.

Operator response
Make sure the LUs are not already in use by another application.

System programmer response
None.

Module
EZAPPOPN

Procedure name
OPCMSG06

EZY0830E  number LUs already in use by another application

Explanation
A connection request has been received for the indicated number of LUs, which are already active.

System action
The system marks the LUs as retryable and will periodically attempt to open them.

Operator response
None.

System programmer response
Correct the startup parameter or the routing file. Also, make sure the specified LU is not already in use by another application. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPOPN

Procedure name
OPCMSG07

EZY0831E  number LUs improperly defined in VTAM definition deck

Explanation
The indicated number of LUs are assigned to the NPF VTAM capture point application and have attributes that are inconsistent with this use.

System action
The program ignores the LUs.

Operator response
Make sure the LUs are not already in use.
**System programmer response**
Correct the startup parameter or the routing file. For more information on the EXEC statement, see z/OS Communications Server: IP Network Print Facility.

**Module**
EZAPPOPN

**Procedure name**
OPCMSG08

**EZY0832E**  SETLOGON failed for name

**Explanation**
An attempt was made to issue a SETLOGON START for an open ACB.

**System action**
The program closes the ACB and will periodically try the OPEN ACB and SETLOGON operations again.

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

**System programmer response**
Make sure enough storage is available for VTAM.

**Module**
EZAPPOPN

**Procedure name**
OPCMSG09

**EZY0833E**  VTAM is currently inactive

**Explanation**
The VTAM support on the system is not active.

**System action**
The program continues to run and will periodically try the VTAM OPEN ACB operation again.

**Operator response**
Make sure VTAM is active before starting Network Print Facility for VTAM.

**System programmer response**
None.

**Module**
EZAPPOPN
VTAM is shutting down

Explanation
The VTAM support on the system is being shut down.

System action
The program continues to run and will periodically try the VTAM OPEN ACB operation again.

Operator response
Reactivate VTAM or end Network Print Facility for VTAM.

System programmer response
None.

Module
EZAPPOPN

No VTAM exists on the system

Explanation
The system contains no VTAM support.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Make sure VTAM is installed before running Network Print Facility for VTAM.

Module
EZAPPOPN

Enter NPFV application operator command

Explanation
This message prompts for an operator command. The possible commands and their usage can be found in the z/OS Communications Server: IP Network Print Facility.
System action
Processing continues.

Operator response
See the z/OS Communications Server: IP Network Print Facility for information on NPFV application operator commands and their usage.

System programmer response
None.

Module
EZAPPAAA, EZAPPCHD

Procedure name
AAIWTOR

EZY0838E Invalid reply specified

Explanation
The reply to message EZY0837I is not correct.

System action
The system prompts for another reply.

Operator response
Re-enter the correct reply.

System programmer response
None.

Module
EZAPPCHD

Procedure name
CHCMSG02

EZY0839E LU specified not in the routing file

Explanation
The LU specified on the LUNAME operator reply was not found in the routing file.

System action
Processing continues.

Operator response
Make sure the correct LUNAME is specified and reenter the request.
System programmer response
Update the routing file if necessary.

Module
EZAPPCLU

Procedure name

EZY0840E  LU specified already open

Explanation
The requested LU was already open.

System action
Processing continues.

Operator response
Make sure that the proper LU was requested.

System programmer response
None.

Module
EZAPPCLU

Procedure name

DOSRCH1

EZY0841E  Routing file record not specified as NPF VTAM record

Explanation
The user requested a new LU be opened by the NPF VTAM capture point application. The one requested is not marked as a VTAM record.

System action
Processing continues.

Operator response
Make sure the correct LU name was entered.

System programmer response
Make sure the routing file has been configured correctly.

Module
EZAPPCLU
Procedure name
BADLU

EZY0842I  LUNAME ADD succeeded

Explanation
The LU was successfully created.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPCHD

Procedure name
OPNLUXIT

EZY0843E  LUNAME ADD failed

Explanation
There was an error while attempting to create a Logical Unit.

System action
Processing continues.

Operator response
Notify the system programmer.

System programmer response
Determine if the appropriate parameters were specified on the EXEC statement for the creation of an LU.

Module
EZAPPCHD

Procedure name
CREATFLD

EZY0844I  LUNAME DEL succeeded

Explanation
A connection has been successfully deleted.
EZY0845E  LUNAME DEL failed - LU does not exist

Explanation
A connection could not be deleted because the specified connection does not exist.

System action
Processing continues.

Operator response
Make sure the connection was properly specified.

System programmer response
None.

Module
EZAPPCHD

Procedure name
n/a

EZY0846E  LUNAME DEL failed - LU still OPEN

Explanation
A connection could not be deleted. The logical unit is still open.

System action
Processing continues.

Operator response
Make sure that the LU is properly specified and not in use by another application.
System programmer response
None.

Module
EZAPPCHD

Procedure name
n/a

**EZY0847I**  Table reload complete

**Explanation**
The end-of-file rules table or the default page format table has been reloaded after the RELOAD command has been issued. For more information, see the z/OS Communications Server: IP Network Print Facility manual.

**System action**
Processing continues.

**Operator response**
None.

System programmer response
None.

Module
EZAPPCHD

Procedure name
FREECOM

**EZY0848E**  LOAD failed for new table

**Explanation**
There was an error loading the new end-of-file rules table or the default page format table. See z/OS Communications Server: IP Network Print Facility for more information.

**System action**
Processing continues.

**Operator response**
Notify the system programmer.

**System programmer response**
Make sure that the table exists and is in the proper library, and that sufficient storage is available to support the table.

Module
EZAPPCHD
**Procedure name**
LOADERR1

**EZY0849E  Format invalid for new table**

**Explanation**
The new end-of-file rules table or the default page format table format was unrecognized. For information on the page formatting macros, see z/OS Communications Server: IP Network Print Facility.

**System action**
Processing continues.

**Operator response**
Notify the system programmer.

**System programmer response**
Correct the table format and recompile and link edit the corrected table.

**Module**
EZAPPCHD

**Procedure name**
LOADERR2

**EZY0850I  Command accepted**

**Explanation**
The requested command has been accepted by the Network Print Facility. This message is issued in response to QUIT, QUIT FORCE, or KILL commands in reply to message EZY0837I, or STOP in reply to message EZY0960I. The command has been accepted as valid and shutdown of the NPF jobs has begun. There might be a significant delay between the issuing of this message and the completion of the shutdown process.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPCHD, EZAPPQSB

**EZY0851E  INVALID INVOCATION OF EZAPPDPE**

**Explanation**
EZAPPDPE is a macro used by EZAPPDPF. This message indicates a logic error in EZAPPDPF. The EZAPPDPF macro is not expanded.
System action
The assembler continues.

Operator response
None.

System programmer response
Call the IBM Software Support Center to report the problem.

Module
EZAPPPFT

Procedure name
EZAPPDPE

EZY0852E INVALID CHARACTER IN field VALUE

Explanation
Only numbers between 0 and 9 can be specified in the value field. The EZAPPDPF macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Correct the value in the specified field of the EZAPPDPF macro. Use only numerics between 0 and 9 in the value field.

Module
EZAPPPFT

Procedure name
EZAPPDPE

EZY0854E PAGE FORMAT NAME IS REQUIRED

Explanation
Each page format you specify with the EZAPPDPF macro in the EZAPPPFT module must have a name. The EZAPPDPF macro is not expanded.

System action
The assembler continues.

Operator response
None.
System programmer response
Specify a name for this page format table entry on the EZAPPDPF macro. For the proper syntax of the macro, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPPFT

Procedure name
EZAPPDPF

**EZY0855E PAGE FORMAT NAME TOO LONG**

**Explanation**
The page format name you specified with the EZAPPDPF macro in the EZAPPPFT module is longer than eight characters. For more information on the EZAPPDPF macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPDPF macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Change the EZAPPDPF macro so the entry name is the correct length. For proper syntax of the macro, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPPFT

Procedure name
EZAPPDPF

**EZY0856E MAX PRES. POSITION GREATER THAN 255**

**Explanation**
The maximum presentation position (MPP) is an optional keyword representing the line length. This can be an integer from 0 to 255. A value of 0 uses the printer’s default of 80 columns. For more information on this keyword, see z/OS Communications Server: IP Network Print Facility. The macro EZAPPDPF is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Change the MPP parameter in the EZAPPDPF macro to a number from 0 to 255.
**EZY0857E**  LEFT MARGIN CANNOT EXCEED MPP

**Explanation**
The maximum presentation position (MPP) represents the line length. The left margin keyword (LM), which identifies where the left margin starts, cannot have a value greater than the value for MPP. A value of 0 defaults to column 1. For more information on the keywords of the EZAPPDPF macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPDPF macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Change the LM parameter in the EZAPPPFPT macro to have a value between 0 and the value for MPP.

---

**EZY0858E**  RIGHT MARGIN < LM OR > MPP

**Explanation**
The keyword representing where the right margin starts has a value that is either less than the value for left margin (LM) or it is greater than the value for the maximum presentation position (MPP). A value of 0 is also permitted and sets the right margin equal to the maximum presentation position (MPP). For more information on the keywords for the EZAPPDPF macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPDPF macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Change the right margin (RM) value to be 0 or less than the value for MPP and greater than the value for LM.
EZAPPDPF

EZY0859E   HT< LM OR > RM

Explanation
The nth value specified for the horizontal tabs (HT=...) is not valid because it is either less than the left margin or greater than the right margin. A value of 0 is also permitted and is ignored. For more information about the EZAPPDPF macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPDPF macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Change the horizontal tabs (HT) value to be 0 or greater than the LM value and less than the RM value.

Module
EZAPPPFT

Procedure name
EZAPPDPF

EZY0860E   MAX PRES. LINE GREATER THAN 255

Explanation
The maximum presentation line (MPL) is an optional keyword representing the page length in lines. This value can be an integer from 0 to 255. A value of 0 uses the printer default of 1 line. For more information on this keyword of the EZAPPDPF macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPDPF macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Change the maximum presentation line (MPL) value to a number between 0 and 255.

Module
EZAPPPFT

Procedure name
EZAPPDPF
EZY0861E  TOP MARGIN CANNOT EXCEED MPL

Explanation
The top margin (TM) keyword specifies what line number the top margin starts on. This value cannot be greater than the value for the maximum page length (MPL). A value of 0 sets the top margin to line 1. For more information on this keyword of the EZAPPDPF macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPDPF macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Change the value for the TM keyword in the EZAPPDPF macro to a number between 0 and the value for MPL.

Module
EZAPPPFT

Procedure name
EZAPPDPF

EZY0862E  BOTTOM MARGIN < TM OR > MPL

Explanation
The value for the bottom margin (BM) cannot be less than the top margin (TM) or greater than the page length (MPL). A value of 0 is also permitted and sets the bottom margin equal to the page length. For more information on this keyword of the EZAPPDPF macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPDPF macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Change the BM value in the EZAPPDPF macro to be 0 or greater than TM and less than MPL.

Module
EZAPPPFT

Procedure name
EZAPPDPF

EZY0863E  VTn < TM OR > BM
Explanation
The nth value specified for vertical tabs (VT= (...)) was not valid. The value must be less than the value for BM (bottom margin) or greater than the value for TM (top margin). A value of 0 is also permitted and is ignored. For more information about the EZAPPDPF macro, see z/OS Communications Server: IP Network Print Facility manual. The EZAPPDPF macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Change the value for the vertical tabs (VT) to be 0 or greater than the value for TM and less than the value for BM.

Module
EZAPPPFT

Procedure name
EZAPPDPF

EZY0864I   MAX PRES. POSITION DEFAULTED TO 80

Explanation
The optional maximum presentation position (MPP) keyword was not included in the EZAPPDPF macro syntax. The MPP will default to a value of 80.

System action
The assembler continues.

Operator response
None.

System programmer response
None.

Module
EZAPPPFT

Procedure name
EZAPPDPF

EZY0865I   LEFT MARGIN DEFAULTED TO 1

Explanation
The left margin (LM) was not specified in the EZAPPDPF macro. The LM will default to 1. To change the left margin, see z/OS Communications Server: IP Network Print Facility under the EZAPPDPF macro.
**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPPFT

**Procedure name**
EZAPPDPF

---

**EZY0866I**  RIGHT MARGIN DEFAULTED TO MPP

**Explanation**
The right margin (RM) was not specified in the EZAPPDPF macro. The RM will default to the value of MPP. For information on how to change the RM, see z/OS Communications Server: IP Network Print Facility.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
None.

---

**Module**
EZAPPPFT

**Procedure name**
EZAPPDPF

---

**EZY0867I**  HTn DEFAULTED TO 0

**Explanation**
The nth horizontal tab parameter was omitted from the list specified with the HT=(ht1,ht2,...) keyword on the EZAPPDPF macro. The nth horizontal tab will default to a value of 0. For information on the HT keyword, see z/OS Communications Server: IP Network Print Facility under the EZAPPDPF macro.

**System action**
The assembler continues.

**Operator response**
None.
EZY0868I  MAX PRES. LINE DEFAULTED TO 1

Explanation
The maximum presentation line (MPL) keyword was not found in the EZAPPDPF macro. MPL will default to a value of 1. The MPL represents the page length in lines.

System action
The assembler continues.

Operator response
None.

System programmer response
None.

EZY0869I  TOP MARGIN DEFAULTED TO 1

Explanation
The optional top margin (TM) keyword was not found in the EZAPPDPF macro. TM will default to a value of 1. For more information on this keyword, see z/OS Communications Server: IP Network Print Facility under the EZAPPDPF macro.

System action
The assembler continues.

Operator response
None.

System programmer response
None.
**Procedure name**
EZAPPDPF

**EZY0870I  BOTTOM MARGIN DEFAULTED TO MPL**

**Explanation**
The bottom margin (BM) was not found in the EZAPPDPF macro. BM will default to a value equal to the value of MPL. For more information on this keyword, see z/OS Communications Server: IP Network Print Facility under the EZAPPDPF macro.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPPFT

**Procedure name**
EZAPPDPF

**EZY0871I  VTn DEFAULTED TO 0**

**Explanation**
The nth vertical tab parameter was omitted from the list specified with the VT=(vt1,vt2,...) keyword on the EZAPPDPF macro. The nth vertical tab will default to a value of 0. For more information on this keyword, see z/OS Communications Server: IP Network Print Facility.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPPFT

**Procedure name**
EZAPPDPF

**EZY0872W  DEFAULT PAGE FORMAT NOT GENERATED**
Explanation
No entry name was specified, or one or more of the following had a non-numeric value as its parameter:
- MPP
- LM
- RM
- HT
- MPL
- TM
- BM
- VT

System action
The assembler continues.

Operator response
None.

System programmer response
Verify that all the keywords in the EZAPPDPF macro have correct values. Also, verify that the format has an entry name. See z/OS Communications Server: IP Network Print Facility for more information on the keywords.

Module
EZAPPPFT

Procedure name
EZAPPDPF

EZY0873E  STB VALUE xxx IS INVALID

Explanation
The only valid settings for the STB operand (Suppress Trailing Blanks) are YES and NO. The EZAPPDPF macro is not expanded.

System action
Processing continues.

Operator response
None.

System programmer response
Correct the value in the STB field of the EZAPPDPF macro.

Module
EZAPPPFT

EZY0874E  TYPE=END MUST BE LAST MACRO
**Explanation**

If the TYPE parameter in the EZAPPEFM macro is END, the macro must mark the end of the table. For information on other possibilities of the TYPE parameter, see z/OS Communications Server: IP Network Print Facility. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Make sure that the EZAPPEFM TYPE=END macro call is the last statement in your EZAPPEFT source deck.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM

---

**EZY0875E**  TYPE PARAMETER IS REQUIRED

**Explanation**
The TYPE keyword is required for identifying the function of the EZAPPEFM macro. The 4 possible values are:
- SYSDFLT
- ENTRY
- PLU
- END

For information on these values, see z/OS Communications Server: IP Network Print Facility manual. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
When coding the EZAPPEFM macro, make sure to include the TYPE keyword and an appropriate value.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM

---

**EZY0876E**  TYPE PARAMETER INVALID
**Explanation**
When specifying the TYPE value when coding the EZAPPEFM macro, use one of the following values:
- SYSDFLT
- ENTRY
- PLU
- END

For more information on these values, see z/OS Communications Server: IP Network Print Facility manual. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Correct the value for the TYPE keyword of the EZAPPEFM macro.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM

**EZY0877E  TYPE=SYSDFLT MUST BE THE FIRST MACRO**

**Explanation**
The SYSDFLT parameter was present for the TYPE keyword when the EZAPPEFM macro was coded. It must be the first EZAPPEFM macro and it must be the only EZAPPEFM macro in the module with the SYSDFLT value. For information on the values of the TYPE keyword, see z/OS Communications Server: IP Network Print Facility. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Change the value of the TYPE value in the EZAPPEFM macro or make this the first EZAPPEFM macro in the module.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM
**EZY0878E  MUST FOLLOW VALID TYPE=ENTRY MACRO**

**Explanation**
Before coding the TYPE=PLU keyword in the EZAPPEFM macro, there must be a TYPE=ENTRY keyword in a previous EZAPPEFM macro. For more information on the TYPE keyword, see z/OS Communications Server: IP Network Print Facility. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Add an EZAPPEFM macro with the TYPE=ENTRY keyword, before the macro with TYPE=PLU.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM

---

**EZY0879E  TABLE ENTRY NAME IS REQUIRED**

**Explanation**
When coding the TYPE=ENTRY keyword in the EZAPPEFM macro, you must specify the name for the end-of-file rules table entry as the label on the macro. For information on the EZAPPEFM macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Include name of the end-of-file rules table entry as the label on the EZAPPEFM macro.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM

---

**EZY0880E  TABLE ENTRY NAME TOO LONG**

---
**Explanation**

When coding the EZAPPEFM TYPE=ENTRY macro, the macro label (name of the table entry) must be between 1 and 8 alphanumeric characters long. For information about the EZAPPEFM macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPEFM macro ends abnormally.

**System action**

The assembler continues.

**Operator response**

None.

**System programmer response**

Change the macro label to contain 8 or less alphanumeric characters.

**Module**

EZAPPEFT

**Procedure name**

EZAPPEFM

**EZY0881E NAME FOUND WHEN NOT TYPE=ENTRY**

**Explanation**

A label on the EZAPPEFM macro is permitted only when the TYPE is ENTRY. See z/OS Communications Server: IP Network Print Facility for more information on the macro. The EZAPPEFM macro is not expanded.

**System action**

The assembler continues.

**Operator response**

None.

**System programmer response**

Correct the EZAPPEFM macro adding the TYPE=ENTRY keyword or deleting the macro label.

**Module**

EZAPPEFT

**Procedure name**

EZAPPEFM

**EZY0882E PLUNAME IS REQUIRED**

**Explanation**

While coding the EZAPPEFM macro, the TYPE=PLU keyword was found and there is no value for the PLUNAME. The EZAPPEFM macro is not expanded.
System action
The assembler continues.

Operator response
None.

System programmer response
Add the PLUNAME keyword and value to the macro. For information on the macro syntax and the PLUNAME keyword, see z/OS Communications Server: IP Network Print Facility.

Module
EZAPPEFT

Procedure name
EZAPPEFM

EZY0883E  PLUNAME FOUND WHEN NOT TYPE=PLU

Explanation
When the TYPE value is not PLU, there must not be a keyword PLUNAME. For information the EZAPPEFM macro, see z/OS Communications Server: IP Network Print Facility. The EZAPPEFM macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Remove the PLUNAME keyword or change the TYPE value to PLU.

Module
EZAPPEFT

Procedure name
EZAPPEFM

EZY0884E  PLUNAME TOO LONG

Explanation
The value for the PLUNAME keyword of the macro syntax is too long. The maximum number of alphanumeric characters that a PLUNAME value can be is 8. For more information, see z/OS Communications Server: IP Network Print Facility. The EZAPPEFM macro is not expanded.

System action
The assembler continues.
Operator response
None.

System programmer response
Change the PLUNAME to be less than or equal to 8 alphanumeric characters.

Module
EZAPPEFT

Procedure name
EZAPPEFM

**EZY0885E  PLUNAME HAS NO SIGNIFICANT CHARs**

Explanation
The PLUNAME value specified for the PLUNAME keyword contained no characters other than * or ?. For naming conventions for the PLUNAME value, see z/OS Communications Server: IP Network Print Facility under the EZAPPEFM macro. The EZAPPEFM macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Change the value specified by the PLUNAME keyword in the EZAPPEFM macro syntax to contain at least 1 significant character.

Module
EZAPPEFT

Procedure name
EZAPPEFM

**EZY0886E  PLUNAME HAS IMBEDDED BLANK OR * **

Explanation
An asterisk or a blank was found in the middle of the PLUNAME value. The asterisk can be used as a wildcard at the beginning or the end of a PLUNAME value, but not in the middle. Blank spaces are not permitted in the middle of a PLUNAME value either. For naming conventions for the PLUNAME value, see the z/OS Communications Server: IP Network Print Facility under the EZAPPEFM macro. The EZAPPEFM macro is not expanded.

System action
The assembler continues.

Operator response
None.
System programmer response
Correct the PLUNAME value in the EZAPPEFM macro.

Module
EZAPPEFT

Procedure name
EZAPPEFM

EZY0887E PLUNAME HAS MIXED * AND ?

Explanation
Both an asterisk and a question mark were found in the PLUNAME value. See the z/OS Communications Server: IP Network Print Facility under the EZAPPEFM macro for information on naming conventions for the PLUNAME value. The EZAPPEFM macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Correct the PLUNAME value in the EZAPPEFM macro.

Module
EZAPPEFT

Procedure name
EZAPPEFM

EZY0888E EOFn= NOT ALLOWED WITH EOF=

Explanation
Both the EOFn and the EOF keywords were found in the syntax for the EZAPPEFM macro. See the z/OS Communications Server: IP Network Print Facility manual for information on the end-of-file rules. The EZAPPEFM macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Delete the appropriate keyword from the macro.
Module
EZAPPEFT

Procedure name
EZAPPEFM

**EZY0889E**  EOF SPEC REQUIRED

**Explanation**
The TYPE value in the EZAPPEFM macro was either SYSDFLT or PLU. With these values, an EOF or EOFn keyword is required. For information on these values, see z/OS Communications Server: IP Network Print Facility under the EZAPPEFM macro. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Add an appropriate EOF or EOFn value to the EZAPPEFM macro.

Module
EZAPPEFT

Procedure name
EZAPPEFM

**EZY0890E**  EOF SPEC NOT ALLOWED ON TYPE=END

**Explanation**
The EOF keywords are not allowed when the TYPE value is END. For more information on the EOF specifications, see z/OS Communications Server: IP Network Print Facility. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Remove the EOF keyword from the EZAPPEFM macro.
EZY0891E  parameter PARAMETER UNRECOGNIZED

Explanation
The value specified for the EOF or EOFn keyword on the EZAPPEFM macro is incorrect. The possible values are:

- EB
- EC
- ES
- STRING
- TIMER

For more information on these values, see z/OS Communications Server: IP Network Print Facility. The EZAPPEFM macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Replace the EOF or EOFn value with the appropriate value.

Module
EZAPPEFT

Procedure name
EZAPPEFM

EZY0892E  EOFn IDLE-BUSY TIMES REQUIRED

Explanation
When entering the value for the EOFn=(TIMER, idleint, busyint) keyword, the TIMER value must be followed by an idle interval and a busy interval, each specified in seconds. For information on these parameters, see z/OS Communications Server: IP Network Print Facility under the EZAPPEFM macro. The EZAPPEFM macro is not expanded.

System action
The assembler continues.

Operator response
None.

System programmer response
Add the idleint and busyint parameters to the value of the EOF or EOFn keyword.

Module
EZAPPEFT
**EZY0893E  EOFn STRING KEEP/DEL REQUIRED**

**Explanation**
If the STRING value is used when coding the end-of-file rules (EOF) keyword for the EZAPPEFM macro, it must be accompanied by a KEEP or DEL parameter. For information on these additions, see z/OS Communications Server: IP Network Print Facility under the EZAPPEFM macro. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Add the KEEP or DEL parameter to the STRING value in the coding of the EOF keyword of the EZAPPEFM macro.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM

---

**EZY0894E  EOFn STRING VALUE REQUIRED**

**Explanation**
When coding the STRING value for the EOF keyword in the EZAPPEFM macro, a parameter must be included. The parameter can either be a character or hexadecimal value that does not exceed 56 characters. For more information, see z/OS Communications Server: IP Network Print Facility under the EZAPPEFM macro. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Correct the macro syntax to contain a parameter for the STRING value of the EOF keyword.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM

---

**EZY0895E  EOFn TIMER NOT ALLOWED WITH LU1**
**Explanation**
The TIMER value can be used with EOF0 or EOF3 and cannot be used with EOF or EOF1. For more information, see z/OS Communications Server: IP Network Print Facility under the EZAPPEFM macro. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Specify separate rules for EOF0, EOF1, and EOF3. Use a rule other than TIMER for the EOF1 keyword of the EZAPPEFM macro.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM

---

**EZY0896E** EOFn STRING DATATYPE NOT C OR X

**Explanation**
When coding the STRING value of the EOF keyword of the EZAPPEFM macro, the parameter must be a hexadecimal string (X) or a character string (C). See z/OS Communications Server: IP Network Print Facility for more information. The EZAPPEFM macro is not expanded.

**System action**
The assembler continues.

**Operator response**
None.

**System programmer response**
Correct the value of the string in the EZAPPEFM macro.

**Module**
EZAPPEFT

**Procedure name**
EZAPPEFM

---

**EZY0897E** EOFn QUOTE REQUIRED AFTER C OR X

**Explanation**
Following the C or X in the STRING parameter, there must be a quotation mark before the parameter. For more information on the parameters of the STRING value, see z/OS Communications Server: IP Network Print Facility under the EZAPPEFM macro. The EZAPPEFM macro is not expanded.
System action
The assembler continues.

Operator response
None.

System programmer response
Correct the macro by adding a single quotation mark before the string and after the X or C in the EZAPPEFM macro. Remember to terminate the string with a single quotation mark also.

Module
EZAPPEFT

Procedure name
EZAPPEFM

EZY0898E   MACRO DID NOT EXPAND

Explanation
The macro syntax was incorrect. This message is displayed with another error message indicating what was not correct.

System action
The assembler continues.

Operator response
None.

System programmer response
Use the message preceding this one to determine the error and correct the macro syntax.

Module
EZAPPEFT

Procedure name
EZAPPEFM

EZY0910I   fssid printerid SYSOUT DYNALLOC/OPEN DONE

Explanation
The creation and opening of the print data set is complete.

System action
Processing continues.

Operator response
None.
System programmer response
None.

Module
EZAPPFD

Procedure name
OPNDDD1

EZY0911I WILL RETRY FOR DYNAALLOC

Explanation
The first attempt to dynamically allocate a data set was not successful.

System action
The program continues and again attempts to dynamically allocate a data set.

Operator response
None.

System programmer response
None.

Module
EZAPPFD

Procedure name
OPNDDD1

EZY0912I fssid printerid JOB jobid jobname IS HELD IN JES QUEUE DUE TO NO VALID DESTINATION

Explanation
Network Print Facility determined there is no valid routing record or options record for the print file. Network Print Facility does not know where to send the print file.

System action
Processing continues.

Operator response
Define a valid routing record or options record, and reissue the print request.

System programmer response
None.

Module
EZAPPFD
**Procedure name**  
OPNDDD1

**EZY0915I**  
`fssid printerid JOB jobid jobname HELD IN JES QUEUE DUE TO NON-ZERO GETREC RC`

**Explanation**  
The Network Print Facility has received an error indication from JES during GETREC processing. No further processing of the output data set can be completed.

**System action**  
The spool file on which the JES GETREC error occurred is changed to a HELD status in the JES queue. Processing continues with the next available print data set.

**Operator response**  
Ensure that the job creating the output has completed successfully. If the output is not printable, it might need to be purged from the JES queue.

**System programmer response**  
Investigate the reason for the nonzero GETREC return code. A previous error message might have been generated, listing a GETREC error condition such as GLRLGE or GLRIOE. These codes are documented in the JES GETREC Processing section of z/OS MVS Using the Functional Subsystem Interface.

**Module**  
EZAPPFA

**EZY0920I**  
`fssid printerid SYSOUT PUT/CLOSE DONE`

**Explanation**  
The creation of a print data set is complete and the close has been performed.

**System action**  
Processing continues.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
EZAPPFD

**Procedure name**  
CLSDDD1

**EZY0921I**  
`fssid printerid SYSOUT SEND DONE`

**Explanation**  
The transmission of the print data set via LPR is complete.
System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFD

Procedure name
CLSDDD1

EZY0922I  fssid printerid SYSOUT RELEASE DONE

Explanation
The JES writer has performed a termination function. This indicates the writer is ending and has successfully closed all files and detached subtasks related to Network Print Facility.

System action
The program ends.

Operator response
None.

System programmer response
None.

Module
EZAPPFD

Procedure name
PFSADD1

EZY0930I  fssid printerid RESET TO FILE START REQUESTED

Explanation
An input record exit has requested the JES writer to reset the file to the beginning for reprocessing. The system restarts the JES output file.

System action
Processing continues.

Operator response
None.
System programmer response
None.

Module
EZAPPFD

Procedure name
SFSDEXIT

**EZY0931E** GETMAIN FOR DEVICE DRIVER SDCB FAILED

Explanation
The JES device driver is unable to allocate its workspace due to lack of virtual storage.

System action
The system ends the JES device driver (writer).

Operator response
Increase the virtual storage assigned to the writer and restart.

System programmer response
None.

Module
EZAPPFD

Procedure name
SFSADD1

**EZY0932E** *ERROR* RC=rc FROM FCM. FSA IS TERMINATING...

Explanation
The JES writer has received a recoverable error response from EZAPPFCM.

System action
The system ends the JES device driver (writer).

Operator response
Look for an accompanying message to provide details of the error. Respond based on the accompanying message.

System programmer response
None.

Module
EZAPPFD
**Procedure name**
SFSDEXIT

**EZY0933E  MINOR ERROR RC=rc FROM FCM IS IGNORED**

**Explanation**
The JES writer has received a recoverable error response from EZAPPFCM such as an incorrect routing.

**System action**
The system continues and processes the next JES output.

**Operator response**
Look for an accompanying message to provide details of the error. Respond based on the accompanying message.

**System programmer response**
None.

**Module**
EZAPPFD

**Procedure name**
SFSDEXIT

**EZY0934E  *ERROR* CAN NOT LOAD EZAPPFCM**

**Explanation**
The JES writer failed in attempting to LOAD EZAPPFCM.

**System action**
The system ends the JES device driver (writer).

**Operator response**
Check the JOBLIB/STEPLIB assignments to make sure that the EZAPPFCM module is accessible. If so, look for indications of a load being unsuccessful from MVS. Correct the error and try again.

**System programmer response**
None.

**Module**
EZAPPFD

**Procedure name**
SFSDEXIT

**EZY0939W  fssid printerid WARNING! Truncating print data record length. Exceeds FSS maximum of 32756.**
**Explanation**
The largest print data set record length for SPIN=DATASET has been exceeded. In order to prevent the FSS from ABENDING, the print data has been truncated.

**System action**
The NPF print data set is created with the maximum record length of 32756 and processing continues.

**Operator response**
Determine that no actual data was lost when the truncation occurred. If automatic truncation is not required, the user can modify the DCB parameters on the DD statement within the job that creates the print data set. The combination that necessitated truncation of data is RECFM=U and BLKSIZE greater than 32756.

Although it might affect other processing in the user's NPF environment, the problem can also be avoided by modifying the FSS to SPIN=GROUP. Make sure that SPIN=GROUP is acceptable before modifying this parameter.

**System programmer response**
None.

**Module**
EZAPPFD

---

**EZY0951E**  
VSAM macro failure on *macro* RETURN CODE = rc REASON CODE = reason

**Explanation**
VSAM has encountered an error processing the Queue file.

**MMM MMM MMM**
The name of the VSAM macro that was unsuccessful.

**RRRR**
The return code provided by VSAM (in)

**CCCC**
The reason code provided by VSAM

**System action**
The program ends abnormally.

**Operator response**
None.

**System programmer response**
See z/OS DFSMS Macro Instructions for Data Sets for an explanation of the reason and response codes.

**Module**
EZAPPQSA

**Procedure name**
QSIMSGPV

---

**EZY0952E**  
NPF Queue Manager: Incorrect parameter/no parameter specified.
**Explanation**
The parameters specified in the PARM field of the JCL EXEC statement are either missing or incorrect.

**System action**
The Queue Manager program ends abnormally.

**Operator response**
Correct the parameters and reissue the EXEC statement. You must specify at least one parameter at startup, the time interval between queue scans. Specify this value in hours, minutes, and seconds format *hhmmss*.

**System programmer response**
None.

**Module**
EZAPPQSA

**Procedure name**
QSIMSG02

---

**EZY0953E** ATTACH for EZAPPQSB failed.

**Explanation**
The ATTACH for the subtask running EZAPPQSB (timer subtask) was unsuccessful. Either the module could not be found or there is insufficient virtual storage to load it.

**System action**
The programs ends abnormally.

**Operator response**
Check the STEPLIB and JOBLIB data sets or the link list for the presence of this module. If it is found, increase the virtual storage and try the job again.

**System programmer response**
None.

**Module**
EZAPPQSA

**Procedure name**
QSIMSG03

---

**EZY0954E** Unable to load EZAPPFCA.

**Explanation**
The LOAD for module EZAPPFCA was unsuccessful. Either the module could not be found or there is insufficient virtual storage to load it.
System action
The program ends abnormally.

Operator response
Check the STEPLIB and JOBLIB data sets or the link list for the presence of this module. If it is found, increase the virtual storage and try the job again.

System programmer response
None.

Module
EZAPPQSA

Procedure name
QSIMSG04

EZY0955E Insufficient storage for program

Explanation
The virtual storage allocation for EZAPPQSA is not sufficient to run the program.

System action
The program ends abnormally.

Operator response
Increase the region size and rerun the program.

System programmer response
None.

Module
EZAPPQSA

Procedure name
QSI00010

EZY0956E DYNALLOC macro failure on ALLOCATION/DEALLOCATION RETURN CODE = rc

Explanation
An error (nonzero value in) was returned by the DYNALLOC macro (SVC 99). rc is the return code.

System action
The program ends abnormally.

Operator response
None.
System programmer response
See z/OS MVS Programming: Authorized Assembler Services Guide for an explanation of the return code.

Module
EZAPPQSA

Procedure name
QSIMSGPD

**EZY0957I** Routing route key operation File filename At time on day day year year

Explanation
Trace output. The routing identified by route key has completed operation operation using file file at the time stated.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPQSA

Procedure name
QSITMMSG

**EZY0958E** Initialization error in Queue Manager

Explanation
An error occurred during initialization in the queue manger. The accompanying message describes the specific error.

System action
Queue manager stops processing.

Operator response
See the accompanying message to determine the cause of the failure.

System programmer response
None.

Module
EZAPPQSA

**EZY0959E** CLOSE failure on Queue File
**Explanation**
The Network Print Facility Queue Manager encountered an error trying to close the QUEUE file while ending. The error message is issued and the Queue Manager continues with ending process.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Look for associated messages and investigate the reason why the QUEUE file might not have been successfully closed.

**Module**
EZAPPQSA

**EZY0960I**  
**NPF Queue Manager: Enter STOP or new time value**

**Explanation**
The operator can reply to this message to change the scan interval or stop the queue manager.

**System action**
The system changes the operation as specified.

**Operator response**
To change the scan interval, reply with a value of hours, minutes, and seconds in the form *hhmmss*. To stop the queue manager, reply with the word *STOP*. To start a trace of the NPF Queue Manager to the system console, reply with the word *TRACE*. To stop tracing the NPF Queue Manager to the system console, reply *NOTRACE*. To change the maximum printer queue depth, reply with a new value from 0-99999 in the form *QDEPTH=nnnnnn*. Specifying *QDEPTH=0* will remove any previously set *QDEPTH* limits and set *DEPTHWTO* to a value of *N*. To request that a WTO be issued when *QDEPTH* is reached, reply with *DEPTHWTO=Y*. *QDEPTH* must not be a nonzero value before *DEPTHWTO* can be turned on. To request that a WTO no longer be issued when *QDEPTH* is reached, reply with *DEPTHWTO=N*.

**System programmer response**
None.

**Module**
EZAPPQSB

**EZY0961I**  
**NPF Queue Manager: Incorrect Reply.**

**Explanation**
The reply to message EZY0960I is not correct.

**System action**
The system reissues message EZY0960I.
Operator response
Respond correctly to the new message.

System programmer response
None.

Module
EZAPPQSB

EZY0962E Unable to Attach EZAPPLPR

Explanation
The queue manager was unable to attach module EZAPPLPR.

System action
The program ends abnormally.

Operator response
None.

System programmer response
Make sure that the module EZAPPLPR is in the proper load library and that enough storage is available to load the module. Increase the region size if necessary.

Module
EZAPPQSA

EZY0963E EZAPPLPR Task Abended Completion Code = cc

Explanation
The task EZAPPLPR has ended abnormally. The reason is indicated by the MVS system completion code displayed in this message. The queue manager might retain the print data set for reprocessing according to user options in effect at the time of the error.

System action
The task ends abnormally.

Operator response
If the error persists, stop queue manager and notify the system programmer.

System programmer response
Use the MVS completion code displayed in this message and the MVS system documentation to determine the cause of the error and respond as indicated.

Module
EZAPPQSA

EZY0964I Queue manager trace turned {on | off}
Explanation
The TRACE for the Queue Manager has been activated (on) or deactivated (off). This message indicates completion of a TRACE or NOTRACE command entered in reply to message EZY0960I.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPQSB

EZY0965I  Queue manager interval changed

Explanation
The interval for Queue Manager scans has been changed in a reply to message EZY0960I.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPQSB

EZY0966I  Queue Manager DEPTHWTO turned off

Explanation
No WTO will be issued when the current QDEPTH limit is reached for any printer.

System action
Processing continues.

Operator response
None.

System programmer response
None.
EZY0967I  Queue Manager DEPTHWTO turned on

Explanation
A WTO will be issued when the current QDEPTH limit has been reached for any printer.

System action
Processing continues.

Operator response
None.

System programmer response
None.

EZY0968I  Queue Manager QDEPTH changed

Explanation
QDEPTH has been changed to the new value specified.

System action
Processing continues.

Operator response
None.

System programmer response
None.

EZY0970I  yy/ddd hh:mm:ss REC data_set_name majorname minorname #recs #dests

Explanation
An entire print file has been received by NPF and queued under the indicated data set name. Fields in the message are as follows:

yy/ddd hh:mm:ss
  Julian date and time.

majorname
  For VTAM, the SLU name. For JES, the jobname.

minorname
  For VTAM, the PLU name. For JES, the username.
#recs
Number of records in the file.

#dests
Number of destinations to which the file will be sent.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPCTR

EZY0971I  yy/ddd hh:mm:ss SND data_set_name printqueue host RC00000 rcdaction datasetaction

Explanation
An attempt to send the print file for the specified data set to LPD for printing has been successful. Fields in the message are as follows:

yy/ddd hh:mm:ss
Julian date and time.

printqueue
The print queue name (truncated to 10 bytes, if necessary).

host
The TCPIP host name or IP address (truncated to 30 bytes, if necessary).

RC00000
Indicates an LPR return code = 0.

Note: RC50xx is a return code presented from LPR. For additional documentation, invoke LPR with the trace option specified against the data set data_set_name referred to in the message.

rcdaction
The following actions can occur:

• QUE (queue) indicates that a queue file record has been created/updated to allow future retries of the send operation by the NPF queue manager.
• DEL (delete) indicates that the queue file record has been deleted.
• QER (queue error) indicates an error occurred while writing a queue file record. The data has been kept to allow future retries of the send operation through manual invocation of LPR. The NPF queue manager is unable to take any further action.

datasetaction
RET (retain) or DEL (delete) indicates whether the corresponding print data set has been kept or deleted.

System action
Processing continues.

Operator response
None.
System programmer response
None.

Module
EZAPPCTR

EZY0972E  yy/ddd hh:mm:ss SND data_set_name printqueue host errcode rcdaction datasetaction

Explanation
An attempt to send the print file for the specified data set to the line printer daemon (LPD) for printing has ended with an error or abend indication.

In the message text:

yy/ddd hh:mm:ss
Julian date and time.

data_set_name
The name of the data set.

printqueue
The print queue name (truncated to 10 bytes, if necessary).

host
The TCPIP host name or IP address (truncated to 30 bytes, if necessary).

errcode
The following types of errors can occur:

• AB00aaa where aaa = system abend code from the line printer control program and spooler (LPR) in hex.
• ATTFAIL indicates that an ATTACH of EZAPPLPR failed.
• RCrrrrr, where rrrrr is 5000 or greater, is a decimal return code generated by LPR; the errcode value is one of the following:
  5011
  See the last EZB0965E message. The data set or file name might be missing from the LPR command.
  5012
  See the last EZB0970E message. Data set organization might be incorrect.
  5013
  See the last EZB0968E message. Unable to allocate data set. Attributes might be incorrect or the data set is inaccessible.
  5014
  See the last EZB0967E message. Unable to access the data set or member.
  5015
  Check for message EZB0969E. LPR's ProcessOperand routine was unable to allocate a data set. Attributes might be incorrect or the data set is inaccessible. The member might not be found.
  5016
  Error detected when processing the LPR options that were passed. Check for the last error messages in the LPR trace.
  5017
  Check for message EZB0923E, EZB0912E, or EZB0914E. The printer name, the host name, or both are missing.
  5018
  Check for message EZB1016E, EZB0953E, EZB0957E, or EZB0959E. LPR did not receive a positive acknowledgment from the print server for the latest command that was sent. One of these messages in the LPR trace will also show the last command that failed to receive the positive acknowledgment.
5019
Check for message EZB0951E. LPR tried to send the control file command, but did not receive a positive acknowledgment.

5020
Check for message EZB0939E or EZB0919E. The error might be caused by an incorrect or unknown userID, HostName, DomainName, or TcpIpServiceName parameter, or one of the parameter values sent in the LPR command, such as the IP address. LPR can also end with this return code if the failure is in the ProcessArguments routine of LPR.

5021
Check for message EZB1026E. The printer name might be unknown or incorrectly defined. Check the printer defined in LPR/NPF or the printer defined to the LPD print server or both.

5022
Check message EZB0940E. LPR is unable to connect to the specified TCP/IP host. The GetHostn call failed to identify the host. If LPR was issued from batch, the IEF142I message indicates Cond Code = 0926 (5022 minus 4096).

5023
Data set larger than 2,147,483,647 bytes.

rcdaction
The following actions can occur:

• QUE (queue) indicates that a queue file record has been created/updated to allow future retries of the send operation by the NPF queue manager.
• DEL (delete) indicates that the queue file record has been deleted.
• QER (queue error) indicates an error occurred while writing a queue file record. The data has been kept to allow future retries of the send operation through manual invocation of LPR. The NPF queue manager is unable to take any further action.

datasetaction
RET (retain) or DEL (delete) indicates whether the corresponding print data set has been kept or deleted.

System action
Processing continues.

Operator response
Contact the system programmer.

System programmer response
If the errcode value is RCrmmm, where rmmm is less than 5000, see the return codes (ERRNOs) information in z/OS Communications Server: IP and SNA Codes for a description of the errors.

If the errcode value is RCrmmm, where rmmm is 5000 or greater, see the errcode description in the explanation for the list of error codes.

If the cause of the error cannot be determined, obtain an LPR trace using the data set name specified in this message to gather more detailed diagnostics.

Module
EZAPPCTR

EZY0973I yy/ddd hh:mm:ss PRG data_set_name printqueue host rcdstate rcdaction datasetaction

Explanation
The NPF queue manager has purged a record from the queue file for one of the following reasons:
• RETAIN(S) retain time expired after a successful transmission.
• RETAIN(U) retain time expired after all retries were exhausted for an unsuccessful transmission.
• A record was marked for deletion by the NPF panel operator.
• The end-of-set (state=X) record was found to be the only remaining record for a multi-destination routing.

The NPF queue manager might also have deleted the print data set specified by the purged queue file record:
• For a single-destination routing, the print data set is deleted when the one queue file record for that routing is purged.
• For a multi-destination routing, the print data set must be retained until all the queue file records for that routing can be processed. Only when the last of these queue file records is purged is the print data set deleted.

Fields in the message are as follows:

`yy/ddd hh:mm:ss`
Julian date and time.

`printqueue`
The print queue name (truncated to 10 bytes, if necessary), or blank (for rcdstate=X only).

`host`
The TCPIP host name or IP address (truncated to 30 bytes, if necessary), or blank (for rcdstate=X only).

`rcdstate`
The state of the queue file record at the time it was purged:
- T - successfully transmitted
- U or R - unsuccessful
- D - deletion requested by operator
- X - end-of-set record for a multi-destination routing

`rcaction`
DEL indicates that the queue file record has been deleted.

`datasetaction`
RET (retain) or DEL (delete) indicates whether the corresponding print data set has been kept or deleted.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPCTR

`EZY0974I` Queue Manager reached QDEPTH for: `printq at host`

**Explanation**
The QDEPTH limit specified has been reached for the printer displayed in the message.

**System action**
Processing continues.
Operator response
Investigate why the printer queue reached its limit. Possible reasons include:

• The printer is not ready to print.
• An unusually large number of print data sets has been sent to the printer within a relatively short period of time.

System programmer response
None.

Module
EZAPPQSA
Chapter 3. EZY1xxx messages

EZY1020E  

hh:mm:ss.th Line linenum: error_description - ‘token’

Explanation
An error was encountered while processing a statement in the configuration data set. The line number of the statement in error is given by linenum. The message includes a description of the error and token identifies the specific text that is in error. Explanations of the errors described by error_description are:

Required valid statement absent
The end of the data set was reached and a valid instance of a required statement was not found. If the statement was present but contained an error, it will have already been flagged by a message.

The missing statement type is identified by token. The required statement types are: TCPIP, LISTENER, and TRANSACTION.

Required parameter missing or incorrect
A required parameter was not specified or was specified incorrectly. The name of the required parameter is identified by token.

The statement is ignored.

Prior statement conflict
The parameters specified on the TRANSACTION statement encountered on line linenum conflict with parameters specified on a previous TRANSACTION statement. The NAME parameter on both statements, identified by token, was the same, but the other parameters were different.

The transaction definition is removed from the set of transactions that can be started by the Listener.

Redeclaration of prior statement
The TRANSACTION statement encountered on line linenum, with NAME token, is the same as a previous TRANSACTION statement.

The current statement is ignored.

No parameter keyword
The parameter token has been specified without an identifying keyword and therefore cannot be recognized. The parameter is ignored.

No parameter value
The parameter token does not specify a value when one is required.

The parameter is ignored.

Unknown statement keyword
The statement type given by token is unknown. Valid statement types are: TCPIP, LISTENER, and TRANSACTION.

The statement is ignored.

Unknown parameter keyword
The parameter token specifies an unknown parameter keyword for this statement type.

The parameter is ignored.

IMPLICIT or EXPLICIT required
The parameter token contains an incorrect value for the transaction type.

The parameter is ignored.

Expected statement
The statement type token was expected on this line but another statement was encountered.

The configuration file is declared incorrect and, after parsing the remaining lines, the Listener ends.
Unexpected statement
The statement type token encountered on this line was unexpected. This type of statement is not valid at this position in the file.

The statement is ignored.

Maximum count exceeded
The current statement is a TRANSACTION statement and this statement has caused the number of these statements to exceed the limit specified by the MAXTRANS parameter of the LISTENER statement. The value that was specified is displayed by message EZY1021I, which is issued after this message.

The statement is ignored.

Value too long
The parameter value specified in token has a length that exceeds the maximum expected length for this parameter. The maximum expected length is displayed by message EZY1021I, which is issued after this message.

The parameter is ignored.

Invalid character
The parameter value specified in token contains a character that is not valid for this parameter such as an alphabetic character where a numeric value is required.

The parameter is ignored.

Value outside valid range
The parameter value in token specifies a numeric value which falls outside the allowed range of values. The allowed maximum and minimum values for this parameter are specified by messages EZY1021I and EZY1022I respectively, which are issued after this message.

The parameter is ignored.

Prior parameter conflict
The parameter value specified in token conflicts with a previous specification of this parameter on this statement.

If the parameter is mandatory then the statement is ignored. Otherwise, all definitions of this parameter are ignored.

Prior conflicting parameters
The parameter token has already been specified more than once on this statement, and these specifications were in conflict with each other.

If the parameter is mandatory, then the statement is ignored. Otherwise, all definitions of this parameter are ignored.

Redeclaration of parameter
The parameter token has already been specified on this statement, but the values are the same.

The statement is valid. This message can be avoided by specifying each parameter only once on a statement.

System action
The entire configuration file is parsed and errors reported. If there are sufficient valid statements to specify all the configuration options, then the Listener continues, otherwise it ends.

Operator response
Edit the configuration file and respecify the statements in error.

System programmer response
Assist the user as required.
Module
EZAIMSPS

Procedure name
varies

EZY1021I  hh:mm:ss.th Maximum: max

Explanation
This message is issued after message EZY1020E to identify the maximum allowed value for the parameter specified in that message.

System action
See message EZY1020E for more information.

Operator response
None.

System programmer response
None.

Module
EZAIMSPS

Procedure name
PRSCFG, ProcessParam

EZY1022I  hh:mm:ss.th Minimum: min

Explanation
This message is issued after message EZY1020E to identify the minimum allowed value for the parameter specified in that message.

System action
See message EZY1021E for more information.

Operator response
None.

System programmer response
None.

Module
EZAIMSPS

Procedure name
ProcessParam

EZY1023E  hh:mm:ss.th File errtype error on ddbname, RC=rc
Explanation

An I/O error occurred while accessing a file with the DDNAME ddname. The I/O activity is given by errtype and is one of the following:

- OPEN
- CLOSE
- GET
- PUT

Additional information about the error can be obtained from rc which has the following values and associated meanings:

1. Internal error.
   Return value for an end of file. Used internally and is not an error. If this value is displayed with this message then a program logic error has occurred.

2. File ABEND exit routine invoked.

3. File SYNAD exit routine invoked.

4. Internal error.
   This return code indicates that a GET or PUT has been performed on a file that has not been opened successfully. If this value is displayed with this message then a program logic error has occurred.

5. The record format for the file is not one of the supported types: FIXED or VARIABLE.

6. Internal error.
   A file structure pointer is not referencing a valid file structure. If this value is displayed with this message then a program error has occurred.

7. Error occurred while closing the file.

8. The ddname is not defined.

System action

The system action depends on errtype as follows:

<table>
<thead>
<tr>
<th>Error</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPEN</td>
<td>The Listener opens the configuration file for input and the message log for output. If either file cannot be opened, the Listener ends.</td>
</tr>
<tr>
<td>CLOSE</td>
<td>Errors associated with closing a file are ignored and processing continues.</td>
</tr>
<tr>
<td>GET</td>
<td>If a GET error occurs before the Listener has been able to retrieve sufficient valid configuration statements to define all the configuration options, the Listener ends. Otherwise processing continues.</td>
</tr>
</tbody>
</table>

Procedure name: main, PRSCFG
Operator response
Take appropriate action based on *errtype* such as specifying the correct ddnames or specifying files with the correct attributes.

System programmer response
Assist the user as required.

Module
EZAIMSPS

Procedure name
See the explanation above.

**EZY1024I**  *event at hh:mm:ss.th on yyddd*

Explanation
One of the following events occurred at the date and time specified.

**Listener invoked**
The Listener has started, and configuration file processing is about to commence.

Procedure name: main

**Listener ended**
The Listener has ended. Termination might have been initiated because of an operator request or because of errors. If an error condition caused the termination, the error will be described in a preceding message.

Procedure name: main

**No available sockets**
The Listener currently has connections open on all the sockets obtained at startup as specified by the MAXACTSKT parameter of the LISTENER statement. The connection that has been requested by a client will be accepted only when a socket becomes available.

Procedure name: ListenProcess

**Security exit rejection**
The user-supplied security exit has rejected a request. Further information about the client that initiated the request is supplied by messages EZY1031W, EZY1032I, and EZY1033I, which are issued after this message.

Procedure name: ReadTRMProcess

**System action**
The system continues.

Operator response
None.

System programmer response
None.

Module
EZAIMSPS
Procedure name
See the explanation above.

**EZY1025W  hh:mm:ss.th description: Code=rc**

**Explanation**
A system error has occurred. The nature of the error is given by description. The error is one of the following:

**No storage available**
A GETMAIN call, which is used to obtain storage for the TRANSACTION statements, has returned an error.
See the GETMAIN macro description for an explanation of the return code rc.

Procedure name: main, ListenProcess, PRSCFG

**Memory deallocation unsuccessful**
A FREEMAIN call has returned an error as evidenced by the return code.
See the FREEMAIN macro description for an explanation of the return code rc.

Procedure name: ReleaseProcess

**CIB free unsuccessful**
The START command CIB was not freed.
No action is required.

Procedure name: main

**Ignored unknown command**
The listener received an unknown command. The command is ignored and processing continues.

Procedure name: main

**System action**
GETMAIN errors cause the Listener to end while FREEMAIN and CIB free errors are ignored as they might occur only during Listener shutdown.

**Operator response**
See the GETMAIN or FREEMAIN documentation and take action as indicated, such as increasing the region size.

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

**Module**
EZAIMSPS

**Procedure name**
See the explanation above.

**EZY1026E  hh:mm:ss.th socket_error, socket_call ERRNO=errno**

**Explanation**
An error occurred during a socket call. The Listener processing at the time of the error and the socket call are given by socket_error and socket_call.

errno is 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.
A description of each of the errors is given below:

**Unable to accept new connection, ACCEPT**
An error occurred when the Listener attempted to complete a connection in response to a connection request from a client.

*Procedure name:* ListenProcess

**Unsuccessful binding to communication port, BIND**
The Listener was unable to bind the port that it is to use for accepting connection requests from clients. The port is specified by the PORT parameter of the LISTENER statement.

*Procedure name:* main

**Socket CLOSE error:**
An error occurred when the Listener attempted to close a connection.

*Procedure name:* ListenProcess, ReleaseProcess

**Error transferring connection to server, GIVESOCKET**
An error occurred when the Listener attempted to pass control of the client connection to the server.

*Procedure name:* CreServPro

**Failed to connect to TCPIP address space, INITAPI**
An error occurred while attempting to connect to the TCPIP address space. The TCPIP address space that is used is specified by the ADDRSPC parameter on the TCPIP statement.

*Procedure name:* main

**Failed setting socket to listen mode, LISTEN**
The LISTEN call, used to enable the socket connected to the listening port for incoming connections, resulted in an error.

*Procedure name:* main

**Socket READ error:**
An error occurred when the Listener attempted to receive data sent from a client program. Further information about the client is provided by message EZY1032I, which is issued after this message.

*Procedure name:* ReadInputProcess, ReadTRMProcess

**Error waiting for communications event, SELECT**
The SELECT call, used to wait for connection requests and data for implicit transactions, has resulted in an error.

*Procedure name:* main

**Failed to obtain a socket to listen on, SOCKET**
The Listener failed to acquire a socket for binding to the listening port.

*Procedure name:* main

**Server failed to take connection, TAKESOCKET**
The Listener issued a GIVESOCKET and the server failed to issue a corresponding TAKESOCKET.

*Note:* ERRNO is not applicable to this error.

**Socket WRITE error:**
An error occurred when the Listener attempted to send data to a client program.

*Procedure name:* WriteRSMProcess

**System action**
The system action depends on the error. If the error occurs during startup processing or during a SELECT call, the Listener ends. Otherwise, the connection that received the error is closed.
Operator response
For errors encountered during startup processing, check that the configuration file TCPIP and LISTENER parameters are correct. Otherwise, proceed based on the value of errno.

System programmer response
Assist the user as required.

Module
EZAIMSPS

Procedure name
See the explanation above.

EZY1027W hh:mm:ss.th Unexpected condition on socket connection

Explanation
An unexpected exception condition or socket closure was detected on a socket connection while receiving data. Further information about the client is provided by message EZY1032I, which is issued after this message.

System action
The connection is closed.

Operator response
None.

System programmer response
None.

Module
EZAIMSPS

Procedure name
ReadInputProcess, SelectResponse, ReadTRMProcess

EZY1028W hh:mm:ss.th description

Explanation
A condition as described by description was detected by the Listener. The conditions are:

Invalid request received
A connection was accepted from a client, but the data received could not be interpreted as a transaction request message (TRM) segment.

The Listener sends a request status message (RSM) segment to the client and then closes the connection.

Procedure name: ReadTRMProcess

Unknown transaction requested
The Transaction Request Message segment from a client specified a transaction that has not been defined to the Listener. Transactions are defined in the configuration file using TRANSACTION statements.

The Listener sends a request status message (RSM) segment to the client and then closes the connection.
**Procedure name:** ReadTRMProcess

**Buffer limit reached**

The amount of data sent by a client for an implicit transaction has exceeded 32KB (including segment lengths and reserved bytes).

The Listener sends a request status message (RSM) segment to the client and then closes the connection.

**Procedure name:** ReadInputProcess

**IMS rollback performed**

The Listener received a nonsuccessful status code when issuing an IMS call to build the IMS message for the requested transaction. The message has been removed by issuing a rollback call. This message will follow another message that has identified the cause of the error.

The Listener sends a request status message (RSM) segment, to the client, if possible, and then closes the connection. Sending the RSM is possible only if the error is detected before the GIVESOCKET call has been issued to pass the socket to the server.

**Procedure name:** CreateServerProcess

**numsktreq** sockets requested, **numsktaloc** sockets allocated

The configuration file specified that **numsktreq** sockets be allocated to process connections; however, only **numsktaloc** sockets were made available by the socket interface. The number of sockets requested is specified by the MAXACTSKT parameter of the LISTENER statement.

**Procedure name:** main

**System action**

The system action is as indicated with each of the above entries.

**Operator response**

See the associated messages, EZY1032I and EZY1033I, which are issued in conjunction with this message to identify the client.

**System programmer response**

Assist the user as required.

**Module**

EZAIMSPS

**Procedure name**

See the explanation above.

```
EZY1029E   hh:mm:ss.th error_description, IMS_function STATUS='status'
```

**Explanation**

The nonsuccessful status code **status** was returned in response to an IMS DC call. A description of the Listener processing at the time of the error and the IMS call function are given by **error_description** and **IMS_function** and are described below. This message appears with messages EZY1032I and EZY1033I, which provide further information about the client.

**Error setting message destination, CHNG**

The Listener attempted to set the destination of an alternate PCB to the transaction code specified in the transaction request message (TRM) segment in preparation for starting the transaction.

**Procedure name:** CreateServerProcess
**Error placing data onto transaction input queue, ISRT**

The error occurred while inserting either the transaction initiation message (TIM) segment or a data segment if it was an implicit transaction.

**Procedure name:** CreateServerProcess

---

**Error committing transaction message, SYNC**

The Listener had successfully inserted the message segments for the requested transaction but encountered the error when issuing the SYNC call.

**Procedure name:** CreateServerProcess

---

**Error rolling back transaction message, ROLB**

The Listener encountered the error on a rollback call it issued to remove a message that it was creating when it encountered an error as described by the above two reasons.

**Procedure name:** CreateServerProcess

---

**System action**

Except for an error caused by a SYNC call, a request status message (RSM) segment indicating an IMS error is sent to the client and the connection is closed. For an error on the SYNC call, the Listener closes the connection. Sending the RSM is possible only if the error is detected before the GIVESOCKET call has been issued to pass the socket to the server. This call is issued just prior to the SYNC call.

---

**Operator response**

None.

---

**System programmer response**

None.

---

**Module**

EZAIMSPS

---

**Procedure name**

See the explanation above.

---

**EZY1030E  hh:mm:ss.th Program Error #xx**

**Explanation**

An internal program error has occurred.

---

**System action**

The Listener ends.

---

**Operator response**

Record the number xx and contact the IBM Software Support Center.

---

**System programmer response**

None.

---

**Module**

EZAIMSPS
**Procedure name**
SelectResp

**EZY1031W  hh:mm:ss.th Security exit RC=rc, Reason=rsn**

**Explanation**
The user-supplied security exit has rejected a request. This message appears with messages EZY1032I and EZY1033I, which provide further information about the client.

**System action**
The Listener sends a request status message (RSM) segment to the client and closes the connection.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAIMSPS

**Procedure name**
ReadTRMProcess

**EZY1032I  hh:mm:ss.th Peer: ipaddress**

**Explanation**
This message appears with other messages to further identify the client by providing the client’s IP address.

**System action**
See other messages.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAIMSPS

**Procedure name**
Various.

**EZY1033I  hh:mm:ss.th Transaction: tran_code**

**Explanation**
This message appears with other messages to further identify the client by printing the transaction code *tran_code* that was requested by the client.
**System action**
See other messages.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAIMSPS

**Procedure name**
Various.

---

**EZY1034I  hh:mm:ss.th AIB error: Retrn=xxxx Reasn=yyyy**

**Explanation**
The Listener detected an error when using the AIB interface to check the status of a requested transaction.

**System action**
The Listener sends a request status message (RSM) segment to the client and closes the connection.

**Operator response**
See the section on AIB INQY call in *IMSA Application Programming: DL/I Calls* for an explanation of the return and reason codes. See messages EZY1032I and EZY1033I for further information about the client.

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

**Module**
EZAIMSPS

**Procedure name**
CreateServerProcess

---

**EZY1035I  hh:mm:ss.th tran_code is unavailable or unknown**

**Explanation**
The transaction *tran_code* that has been requested by a client is currently unavailable or is unknown to IMS. See messages EZY1032I and EZY1033I for further information about the client.

**System action**
The Listener sends a request status message (RSM) segment to the client and closes the connection.

**Operator response**
Make sure that a client program requests a transaction that has been defined to IMS.
System programmer response
Assist the user as required.

Module
EZAIMSPS

Procedure name
CreateServerProcess

EZY1120E  No TYPE specified

Explanation
The EZASMI macro requires a TYPE parameter. None was specified.

System action
The system ends macro generation.

Operator response
Specify a TYPE parameter on the EZASMI macro.

System programmer response
None.

Module
EZASMI

Procedure name
.TYP0

EZY1121E  TYPE parameter is invalid

Explanation
The TYPE parameter specified for this macro is not valid.

System action
The system ends macro generation.

Operator response
Specify a valid TYPE for the macro and reissue the macro. See the z/OS Communications Server: IP Programmer's Guide and Reference for information about the macro and values for the TYPE parameter.

System programmer response
None.

Module
EZASMI
**Procedure name**
.TYP42

**EZY1122E**  PPPP parameter required for type TTTT

**Explanation**
The parameter PPPP is required for the EZASMI macro when type TTTT is specified.

**System action**
The system ends macro generation.

**Operator response**
See z/OS Communications Server: IP Programmer's Guide and Reference to determine the valid parameters for the macro type you specified.

**System programmer response**
None.

**Module**
EZASMI

**Procedure name**
.PG01L1, .PG03L1, .PG04L1, .PG05L1, .PG06L1, .PG07L1, .PG08L1, .PG08OKC, .PG08OKD, .PG08END, .PG09L1, .PG10L1, .PG11L1, .PG11OKD, .PG13L1, .PG13OKB, .PG14L1, .PG15L1, .PG15OKB, .PG18L1, .PG19L1, .PG19OKD, .PG20L1, .PG21A2, .PG21A3

**EZY1123W**  PPPP parameter invalid for type TTTT

**Explanation**
The parameter PPPP is not required for the EZASMI macro when type TTTT is specified. It is ignored.

**System action**
Macro generation continues.

**Operator response**
See z/OS Communications Server: IP Programmer's Guide and Reference to determine the valid parameters for the macro type you specified.

**System programmer response**
None.

**Module**
EZASMI

**Procedure name**
**EZY1124E**  xRETMSK Required if xSNDMSK specified

**Explanation**
On the SELECT, if a send mask is specified for read, write, or exception, the return mask must be specified for the same function.

**System action**
The system ends macro generation.

**Operator response**
Specify the return mask for the read, write, or exception function for which a send mask was specified.

**System programmer response**
None.

**Module**
EZASMI

**Procedure name**
.SELF4, .SELG4, .SELH4

---

**EZY1125E**  Invalid AF specified, INET assumed

**Explanation**
A literal specified for AF on the SOCKET function is not valid. The macro assumed AF=INET.

**System action**
The system ends macro generation.

**Operator response**
The literal for AF on the SOCKET function must be either INET or IUCV.

**System programmer response**
None.

**Module**
EZASMI

**Procedure name**
.SOCA01

---

**EZY1126E**  ECB not permitted for INITAPI

**Explanation**
The ECB parameter is not permitted for TYPE=INITAPI.

**System action**
The system ends macro generation.
**Operator response**
Remove the ECB parameter and reissue the macro.

**System programmer response**
None.

**Module**
EZASMI

**Procedure name**
.INIO

---

**EZY1127E**  
Invalid literal specified for PPPP

**Explanation**
The literal specified for parameter PPPP is not a valid value.

**System action**
The system ends macro generation.

**Operator response**
See z/OS Communications Server: IP Programmer's Guide and Reference for the valid parameter values.

**System programmer response**
None.

**Module**
EZASMI

**Procedure name**
.FCNCO1, .GSON05, .IOCC0E, .RCFF1, .SNDF1, .SDBT01, .SSON05

---

**EZY1128E**  
Invalid socket type specified, STREAM assumed.

**Explanation**
The socket type specified for a TYPE=SOCKET macro is not valid. SOCTYPE=STREAM is assumed.

**System action**
The system ends macro generation.

**Operator response**
Valid values for the SOCTYPE parameter are STREAM, DATAGRAM, and RAW.

**System programmer response**
None.
Module
EZASMI

Procedure name
.SOCCCT3

**EZY1129E** No ECB specified for SYNC

**Explanation**
The TYPE=SYNC requires an ECB parameter.

**System action**
The system ends macro generation.

**Operator response**
Specify an ECB parameter on the TYPE=SYNC request and reissue the macro.

**System programmer response**
None.

Module
EZASMI

Procedure name
.ECB1

**EZY1130E** Invalid value for *label*

**Explanation**
The label specified is not a valid assembler label.

**System action**
The system ends macro generation.

**Operator response**
Specify a label that is a maximum of eight characters in length.

**System programmer response**
None.

Module
EZASMI

Procedure name
.N0003

**EZY1218E** mm/dd/yy hh:mm:ss PROGRAM programname DISABLED TRANID= transactionid PARTNER INET ADDR=inetaddress PORT=portnumber
**Explanation**
The Listener checked the status of the program associated with the transaction. It was not enabled.

- `mm/dd/yy` is the date (month/day/year) of the message.
- `hh:mm:ss` is the time (hours:minutes:seconds) of the message.

- `programname` is the name of the program that is associated with the transaction requested by the connecting client.
- `transactionid` is the name of the transaction that was requested by the connecting client.
- `inetaddress` is the internet address of the connecting client.
- `portnumber` is the connecting client's port number.

**System action**
Listener continues.

**Operator response**
Use CEMT to determine and correct the status of the program.

**System programmer response**
None.

**Module**
EZACIC02

**Procedure name**
LISTENER

**EZY1219E**  

- `mm/dd/yy` is the date (month/day/year) of the message.
- `hh:mm:ss` is the time (hours:minutes:seconds) of the message.
- `eventtype` is the type of event: READ, WRITE, or EXCEPTION.
- `transactionid` is the name of the Listener's CICS transaction.
- `ipaddress` is the remote IP address of the client.
- `portnumber` is the remote port number of the client.

**Explanation**
The CICS Listener was notified about an unexpected event.

- `mm/dd/yy` is the date (month/day/year) of the message.
- `hh:mm:ss` is the time (hours:minutes:seconds) of the message.

- `eventtype` is the type of event: READ, WRITE, or EXCEPTION.
- `transactionid` is the name of the Listener's CICS transaction.
- `ipaddress` is the remote IP address of the client.
- `portnumber` is the remote port number of the client.

**System action**
The Listener closes the connection and continues processing.

**Operator response**
Contact the system programmer.
System programmer response

If the event type is EXCEPTION, investigate whether or not the client is attempting to send out-of-band data. If necessary, have the client avoid sending out-of-band data. If the event type is not EXCEPTION or the client is not attempting to send out-of-band data, then contact the IBM Software Support Center.

Module

EZACIC02

Procedure name

LISTENER

EZY1220E  

**mm/dd/yy hh:mm:ss READ FAILURE ON CONFIGURATION FILE PHASE=phase**

**EIBRESP2=response**

Explanation

EZACIC21 was unable to read the IP CICS Sockets configuration file, EZACONFG.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

*phase* is the IP CICS Sockets initialization phase.

*response* is the response from CICS when reading the IP CICS Sockets configuration file.

System action

If the ABEND code is AEXY, then the listener ends normally. Otherwise, the listener ends with an ABEND code of EZAL.

Operator response

Notify the CICS system programmer.

System programmer response

Use the EIBRESP2 value to determine the problem and correct the file. See [http://www.ibm.com/software/htp/cics/library/](http://www.ibm.com/software/htp/cics/library/) for information about EIBRESP2 values. If the EIBRESP2 value is zero, then the EZACONFG file has been defined as remote. If this is the configuration file you want, then verify that no CICS Sockets programs can run directly in the file owning region. This can cause the file to become disabled. Ensure that EZACIC20 is not in the file owning region PLT, and that the EZAC and EZAO transactions are unable to run directly in the file owning region. Attempts to open the file will fail if the file is defined with a value of YES specified in the ADD, DELETE, or UPDATE parameters in the CICS file definition in more than one CICS region.

Module

EZACIC21

Procedure name

INITIALIZATION

EZY1221E  

**mm/dd/yy hh:mm:ss CICS SOCKETS ENABLE FAILURE EIBRCODE BYTE2 = resp_code**

Explanation

The attempt to enable the task related user exit (TRUE) failed.

*mm/dd/yy* is the date (month/day/year) of the message.
hh:mm:ss is the time (hours:minutes:seconds) of the message.

resp_code is the CICS response code from attempting to enable IP CICS Sockets Task Related User Exit (TRUE).

**System action**
Terminate the transaction.

**Operator response**
Notify the CICS system programmer.

**System programmer response**
Use the EIBRESP2 value to determine the problem and correct the file. An EIBRCODE BYTE2 value of 20 indicates the TRUE is already enabled. This will occur if you disable the interface using EZAO,STOP,CICS transaction and then immediately issue EZAO,START,CICS transaction before the Task Related User Exit (TRUE) is completely disabled from the previous EZAO,STOP,CICS transaction. See [http://www.ibm.com/software/htp/cics/library/](http://www.ibm.com/software/htp/cics/library/) for information about EIBRCODEs.

**Module**
EZACIC21

**Procedure name**
INITIALIZATION

EZY1222E  

**mm/dd/yy hh:mm:ss**  
CICS/SOCKETS REGISTRATION FAILURE RETURN code= return_code

**Explanation**
The attempt to register the CICS Sockets Feature to z/OS failed.

**System action**
Terminate the transaction.

**Operator response**
Contact your System Administrator.

**System programmer response**
See the z/OS MVS Programming: Product Registration for information about the values for return_code.

**Module**
EZACIC21

**Procedure name**
INITIALIZATION

EZY1223E  

**mm/dd/yy hh:mm:ss**  
CICS/SOCKETS ATTACH FAILURE RETURN CODE = return_code REASON CODE = reason_code

**Explanation**
An attempt to attach one of the pool subtasks failed.
**System action**
Stop attaching pool subtasks. The size of the pool is determined by the number of subtasks successfully attached.

**Operator response**
Contact the CICS system programmer.

**System programmer response**
See the z/OS MVS Programming: Authorized Assembler Services Reference ALE-DYN for information about the values for `return_code` and `reason_code` and make appropriate adjustments to your CICS environment.

**Module**
EZACIC21

**Procedure name**
INITIALIZATION

**EZY1224I**  
`mm/dd/yy hh:mm:ss CICS/ SOCKETS INITIALIZATION SUCCESSFUL USING tasking_method`

**Explanation**
The CICS socket interface has completed initialization successfully.

- `mm/dd/yy` is the date (month/day/year) of the message.
- `hh:mm:ss` is the time (hours:minutes:seconds) of the message.

`tasking_method` is the tasking method used to support the EZASOKET calls. The possible methods are:

**Reusable MVS subtasks**
Signifies that the IP CICS socket interface is using MVS subtasks from the pool generated according to the value specified on the NTASKS configuration parameter.

**Non-reusable MVS subtasks**
Signifies that the IP CICS socket interface is attaching an MVS subtask for each IP CICS Sockets-enabled application because NTASKS=0.

**Open Transaction Environment**
Signifies that the IP CICS socket interface is enabled to use CICS Open Transaction Environment. All EZASOKET calls will be processed on an Open API, L8, TCB. Programs calling EZASOKET should be coded to threadsafe programming standards and defined to CICS as CONCURRENCY(THREADSAFE) to benefit from this environment.

**System action**
Continue with execution.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZACIC21
Explanation
The STARTBR command used for the configuration file has failed.

System action
Terminate the transaction.

Operator response
Contact the CICS system programmer.

System programmer response
Use the EIBRESP2 value to determine the problem. Check the CICS definition of the Configuration file to ensure the browse operation is permitted. See http://www.ibm.com/software/htp/cics/library/ for information about EIBRESP2 values.

Module
EZACIC21

Procedure name
INITIALIZATION

Explanation
The READNEXT command used for the configuration file has failed.

System action
Terminate the transaction.

Operator response
Contact the CICS system programmer.

System programmer response
Use the EIBRESP2 value to determine the problem. Check the CICS definition of the Configuration file to ensure the browse operation is permitted. See http://www.ibm.com/software/htp/cics/library/ for information about EIBRESP2 values.

Module
EZACIC21

Procedure name
INITIALIZATION
**EZY1227E  mm/dd/yy hh:mm:ss CICS/SOCKETS INVALID LISTENER TRANID = tran**

**Explanation**
The Listener transaction `tran` was not defined to CICS.

**System action**
Terminate Listener Initialization.

**Operator response**
Use CICS facilities to define the Listener transaction and program. Then use EZAO to start the Listener.

**System programmer response**
None.

**Module**
EZACIC21

**Procedure name**
INITIALIZATION

**EZY1228E  mm/dd/yy hh:mm:ss CICS/SOCKETS LISTENER TRANSACTION tran DISABLED**

**Explanation**
The Listener transaction `tran` could not be started because it was disabled.

**System action**
Terminate Listener Initialization.

**Operator response**
Use CICS facilities to enable the transaction and then start the Listener using EZAO.

**System programmer response**
None.

**Module**
EZACIC21

**Procedure name**
INITIALIZATION

**EZY1229E  mm/dd/yy hh:mm:ss CICS SOCKETS LISTENER TRANSACTION tran NOT AUTHORIZED**

**Explanation**
The Listener transaction `tran` could not be started because it was not authorized.

**System action**
Terminate Listener Initialization.
**Operator response**
Use CICS facilities to authorize starting the Listener transaction and then start the Listener using EZAO.

**System programmer response**
None.

**Module**
EZACIC21

**Procedure name**
INITIALIZATION

---

**EZY1246E mm/dd/yy hh:mm:ss CICS SOCKETS LISTENER PROGRAM ID mmmmmmmmm INVALID**

**Explanation**
The Listener transaction could not be started because program mmmmmmmmm is not defined.

**System action**
Terminate Listener Initialization.

**Operator response**
If the program ID is correct, use CICS facilities to define it. If it is not correct, use the EZAC transaction to correct the CICS Sockets Configuration file.

**System programmer response**
None.

**Module**
EZACIC21

**Procedure name**
INITIALIZATION

---

**EZY1247E mm/dd/yy hh:mm:ss CICS SOCKETS LISTENER PROGRAM ID mmmmmmmmm DISABLED**

**Explanation**
The Listener transaction could not be started because program mmmmmmmmm is disabled.

**System action**
Terminate Listener Initialization.

**Operator response**
Use CICS facilities to enable the program and then use EZAO to start the Listener.

**System programmer response**
None.
Module
EZACIC21

Procedure name
INITIALIZATION

**EZY1250E**  
`mm/dd/yy hh:mm:ss CICS/SOCKETS LISTENER tran NOT ON CONFIGURATION FILE`

**Explanation**
The Listener transaction `tran` is not defined on the CICS Sockets configuration file.

**System action**
Terminate Listener Initialization.

**Operator response**
If the Listener transaction name is correct, use the EZAC transaction to define it on the CICS Configuration file. If the name is not correct, correct it on the EZAO transaction.

**System programmer response**
None.

Module
EZACIC21

Procedure name
INITIALIZATION

**EZY1251E**  
`mm/dd/yy hh:mm:ss CICS SOCKETS MODULE mmmmmmm ABEND xxxx`

**Explanation**
The CICS Sockets module `mmmmmmmm` has abended.

**System action**
Terminate the transaction.

**Operator response**
Contact the IBM Software Support Center.

**System programmer response**
None.

Module
EZACIC21

Procedure name
INITIALIZATION

**EZY1252E**  
`mm/dd/yy hh:mm:ss UNABLE TO LOAD EZASOH03 ERROR CODE= error_code REASON CODE= reason_code`
Explanation
During CICS Sockets initialization, the attempt to load module EZASOH03 failed.

System action
Terminate Initialization.

Operator response
Contact the CICS system programmer.

System programmer response
See the z/OS MVS Programming: Authorized Assembler Services Reference LLA-SDU for information about the values for error_code and reason_code to determine why the module would not load. Also, look for associated MVS messages.

Module
EZACIC21

EZY1253E  mm/dd/yy hh:mm:ss CICS/ SOCKETS LISTENER tran NOT ON CONFIGURATION FILE

Explanation
An EZAO STOP LISTENER transaction was run with an invalid Listener name.

System action
Present the panel to correct the name.

Operator response
Correct the name and retry termination.

System programmer response
None.

Module
EZACIC22

Procedure name
TERMINATION

EZY1254E  mm/dd/yy hh:mm:ss CACHE FILE ERROR RESP2 VALUE ****** CALL # *

Explanation
An error occurred on a cache file operation.

System action
Return to the calling program with an error response.

Operator response
Contact the CICS system programmer.
System programmer response
Use the RESP2 value to determine the error and correct the cache file. See http://www.ibm.com/software/htp/cics/library/ for information about RESP2 values.

Module
EZACIC25

Procedure name
DOMAIN NAME SERVER FUNCTION

EZY1255E  mm/dd/yy hh:mm:ss TEMPORARY STORAGE ERROR RESP2 VALUE ***** CALL # *

Explanation
An error occurred on a temporary storage operation in EZACIC25.

System action
Return to the calling program with an error response.

Operator response
Use the RESP2 value to determine the error. Contact the IBM Software Support Center. See http://www.ibm.com/software/htp/cics/library/ for information about RESP2 values.

System programmer response
None.

Module
EZACIC25

Procedure name
DOMAIN NAME SERVER FUNCTION

EZY1256E  mm/dd/yy hh:mm:ss CICS SOCKETS INTERFACE NOT ENABLED PRIOR TO LISTENER STARTUP

Explanation
An attempt to start a Listener was made when the CICS socket interface was inactive.

System action
Return error and terminate transaction EZAO.

Operator response
Use transaction EZAO to start the CICS socket interface prior to starting the Listener.

System programmer response
None.

Module
EZACIC21
Procedure name
INITIALIZATION

EZY1258I  module ENTRY POINT IS address

Explanation
This message displays the entry point address of a module.
module is the name of the module.
address is the entry point address of the module.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZACIC01, EZACIC02

EZY1259E  mm/dd/yy hh:mm:ss IOCTL CALL FAILURE TRANSACTION=transactionid
          TASKID=tasknumber ERRNO=errno

Explanation
Listener transaction transactionid experienced a failure on the IOCTL call.

In the message text:

mm/dd/yy
  The date (month/day/year) of the message.

hh:mm:ss
  The time (hours:minutes:seconds) of the message.

transactionid
  The name of the transaction under which the Listener is executing.

tasknumber
  The CICS task number of the Listener task.

errno
  errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets
  extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
If the error is during initialization of the Listener, then the Listener transaction transactionid terminates.
Otherwise, the Listener closes the socket that was being processed and resumes normal processing.

Operator response
Use the errno value to determine the cause of the failure.
System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

**EZY1260E**  

mm/dd/yy hh:mm:ss  
EZACIC03 ATTACH FAILED GPR15=xxxxxxx  
ERRNO=errno  
TRAN=tran  
TASK=cicstask

Explanation
An ATTACH for an MVS subtask has failed. The reason code is in GPR 15.  

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The task related user exit (TRUE) for this transaction is disabled. The transaction abends with an AEY9.

Operator response
Contact the CICS system programmer.

System programmer response
Determine the cause for the ATTACH failure and correct.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

**EZY1261I**  

mm/dd/yy hh:mm:ss  
EZACIC03 ATTACH SUCCESSFUL, TCB ADDRESS= tcbaddr  
TERM=term  
TRAN=tran  
TASK=cicstask

Explanation
An ATTACH for an MVS subtask was successful. This message is produced only for Listeners and for those tasks that cannot be accommodated within the pool of reusable tasks.

**Result:** If you specify the character L as the last character in the subtask ID parameter of an INITAPI socket command, then the IP CICS Socket task related user exit (TRUE) assumes that the CICS transaction is a listener causing the TRUE to attach a new task to support the listener's socket commands.

mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

tcbaddr is the address of the Task Control Block (TCB) being attached.

term is the CICS terminal ID associated with the CICS transaction identified by tran.

tran is the name of the CICS transaction that was requested.

cicstask is the task number of the CICS transaction identified by tran.
System action
Processing continues.

Operator response
If this message happens frequently, increase the size of the reusable task pool, NTASKS, for this CICS. Increasing NTASKS appropriately will prevent overhead incurred with attaching the subtask. See the EZACICD TYPE parameter in z/OS Communications Server: IP CICS Sockets Guide for information the NTASKS value.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1262E  mm/dd/yy hh:mm:ss GWA ADDRESS INVALID UEPGAA=xxxxxxxx TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) detected an invalid GWA address.

System action
The TRUE is disabled and the task abends with an AEY9.

Operator response
Use EZAO to stop (immediate) and start the CICS socket interface. If the problem repeats, contact the IBM Software Support Center.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1263E  mm/dd/yy hh:mm:ss TIE ADDRESS INVALID UEPGAA=xxxxxxxx TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) detected an invalid TIE address.

System action
The TRUE is disabled and the task abends with an AEY9.
Operator response
Use EZAO to stop (immediate) and start the CICS socket interface. If the problem repeats, contact the IBM Software Support Center.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1264E  mm/dd/yy hh:mm:ss FLAG WORD ADDRESS INVALID UEPFLAGS= xxxxxxxx ERRNO=errno
TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) detected an invalid flag word address.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.

Operator response
Use EZAO to stop (immediate) and start the CICS socket interface. If the problem repeats, contact the IBM Software Support Center.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1265E  mm/dd/yy hh:mm:ss CICS VERSION UNSUPPORTED GWACIVRM=xxxxx ERRNO=errno
TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) detected a version of CICS which it does not support. The CICS version must be 3 or above.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.
Operator response
Contact the CICS system programmer.

System programmer response
The CICS socket interface requires CICS V3R3 or later.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1267E mm/dd/yy hh:mm:ss ROUTING TASK FUNCTION INVALID UERTIFD=xx ERRNO=errno
TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) detected an invalid routing task function.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.

Operator response
If this happens repeatedly, use EZAO to STOP (immediate) the CICS socket interface and then START it. If it still happens, contact the IBM Software Support Center.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1268E mm/dd/yy hh:mm:ss SAVE AREA ADDRESS INVALID UEPHSMA= xxxxxxxx ERRNO=errno
TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) detected an invalid save area address.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.
Operator response
Contact the IBM Software Support Center.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1269E  mm/dd/yy hh:mm:ss PARM LIST ADDRESS INVALID GPR1= xxxxxxxx ERRNO=errno
TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) detected an invalid parameter list on a call request from the CICS application program.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.

Operator response
Check the application program calls to the CICS socket interface to ensure that each call has the correct number and type of parameters.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1270E  mm/dd/yy hh:mm:ss PARM nn ADDRESS INVALID ADDRESS= xxxxxxxx ERRNO=errno
TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) detected an invalid parameter address on a call request from the CICS application program. nn is the number of the parameter.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.
Operator response
Check the application program calls to the CICS socket interface to ensure that the parameter addresses are valid (not zero). This problem is most common in assembler language and C applications.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1271E  mm/dd/yy hh:mm:ss TOKERR=xxxxxxx ERRNO=errno TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) detected a token error on an internal token used to coordinate CICS transaction activity with TCP/IP activity.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.

Operator response
Contact the IBM Software Support Center.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1272E  mm/dd/yy hh:mm:ss INVALID SOCKET/FUNCTION CALL FUNCTION= xxxx ERRNO=errno TRAN=tran TASK=cicstask

Explanation
A call to EZASOKET specified in invalid function.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.
Operator response
Correct the call and try again.

System programmer response
None.

Module
EZACIC01

Procedure name
task related user exit (TRUE)

EZY1273E  mm/dd/yy hh:mm:ss IUCV SOCK/FUNC TABLE INVALID FUNCTION= xxxx ERRNO=errno
TRAN=tran TASK=cicstask

Explanation
A call to EZACICAL specified a function that was not valid.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.

Operator response
Correct the call and try again.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1274E  mm/dd/yy hh:mm:ss INCORRECT EZASOKET PARM COUNT FUNCTION= xxxx ERRNO=errno
TRAN=tran TASK=cicstask

Explanation
A call to EZASOKET specified invalid number of parameters.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE is disabled and the task abends with an AEY9.

Operator response
Correct the call and try again.
**System programmer response**
None.

**Module**
EZACIC01

**Procedure name**
TASK RELATED USER EXIT (TRUE)

---

**EZY1275E**

*mm/dd/yy hh:mm:ss MONITOR CALLS NOT SUPPORTED UERTFID=xx ERRNO=errno*

*TRAN=tran TASK=cicstask*

**Explanation**
The task related user exit (TRUE) detected a monitor call which is not supported for this version of CICS. 
*errno* is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

**System action**
The TRUE is disabled and the task abends with an AEY9.

**Operator response**
Contact the IBM Software Support Center.

---

**System programmer response**
None.

**Module**
EZACIC01

**Procedure name**
TASK RELATED USER EXIT (TRUE)

---

**EZY1276E**

*mm/dd/yy hh:mm:ss EDF CALLS NOT SUPPORTED UERTFID=xx ERRNO=errno TRAN=tran TASK=cicstask*

**Explanation**
The task related user exit (TRUE) detected an EDF (Execute Diagnostic Facility) call. This TRUE does not support EDF calls. 
*errno* is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

**System action**
The TRUE is disabled and the task abends with an AEY9.

**Operator response**
Contact the IBM Software Support Center.
System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1277I mm/dd/yy hh:mm:ss EZACIC03 DETACHED TCB ADDRESS=xxxxxxxx ERRNO=errno
TRAN=tran TASK=cicstask

Explanation
An attached subtask is terminating.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets
extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The TRUE detaches the MVS subtask.

Operator response
None.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1278I mm/dd/yy hh:mm:ss EZACIC03 DETACH SUCCESSFUL TCB ADDRESS=xxxxxxxx TRAN=tran
TASK=cicstask

Explanation
An attached subtask is terminating.

System action
The TRUE detaches the MVS subtask.

Operator response
None.

System programmer response
None.
Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1279E  mm/dd/yy hh:mm:ss INVALID SYNC PT COMMAND DISP=xx TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) Detected an invalid Sync Point command.

System action
Disable the TRUE and return to the caller.

Operator response
Contact the IBM Software Support Center.

System programmer response
None.

Module
EZACIC01

Procedure name
TASK RELATED USER EXIT (TRUE)

EZY1280E  mm/dd/yy hh:mm:ss INVALID RESYNC COMMAND DISP=xx TRAN=tran TASK=cicstask

Explanation
The task related user exit (TRUE) Detected an invalid Resync command.

System action
Disable the TRUE and return to the caller.

Operator response
Contact the IBM Software Support Center.

System programmer response
None.

Module
EZACIC01

EZY1282E  mm/dd/yy hh:mm:ss 10999 ABEND reasonxx

Explanation
The ESTAE processing in EZACIC03 could not be completed because of reasonxx.
System action
Allow the ABEND to percolate.

Operator response

System programmer response
None.

Module
EZACIC03

Procedure name
MVS SUBTASK

EZY1285E mm/dd/yy hh:mm:ss CICS/ SOCKETS LISTENER TRANSACTION tran NOT ON CONFIGURATION FILE

Explanation
The Listener attempting to start does not have a description record on the CICS Sockets configuration file.

System action
Listener terminates.

Operator response
Contact CICS system programmer.

System programmer response
Add the Listener to the configuration file using EZAC and try again.

Module
EZACIC02

Procedure name
LISTENER

EZY1286E mm/dd/yy hh:mm:ss READ FAILURE ON CICS/ SOCKETS CONFIGURATION FILE TRANSACTION= tran EIBRESP2= rrrrr

Explanation
The Listener could not read the configuration file.

System action
Listener terminates.

Operator response
Contact CICS system programmer.
System programmer response
Use the CICS APR to interpret the value of EIBRESP2. If the file is not known to CICS, perform the installation steps for the configuration file.

Module
EZACIC02

Procedure name
LISTENER

EZY1287E

mm/dd/yy hh:mm:ss EZYCIC02 GETMAIN FAILURE FOR VARIABLE STORAGE TRANSACTION=tran EIBRESP2=rrrrr

Explanation
EZACIC02 could not obtain the variable storage it requires to execute.

System action
Listener terminates.

Operator response
Contact CICS system programmer.

System programmer response
Use the CICS APR to interpret the value of EIBRESP2. Correct your CICS configuration as indicated.

Module
EZACIC02

Procedure name
LISTENER

EZY1288E

mm/dd/yy hh:mm:ss CICS SOCKETS MODULE mmmmmmmmm ABEND aaaa

Explanation
An abend has occurred in module mmmmmmmmm of the CICS socket interface.

System action
Listener terminates.

Operator response

System programmer response
None.
Module
EZACIC02

Procedure name
LISTENER

EZY1289I mm/dd/yy hh:mm:ss CICS LISTENER TRANSACTION tran taskno TERMINATING

Explanation
The Listener is ending. This could be a normal shutdown situation or a failure related to the Listener socket. If it is the latter, a previous message described the failure.

In the message text:

mm/dd/yy
The date (month/day/year) of the message.

hh:mm:ss
The time (hours:minutes:seconds) of the message.

tran
The listener transaction ID.

taskno
The CICS task number assigned to the listener transaction ID.

System action
The Listener ends.

Operator response
None.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CICS Listener

Module
EZACIC02

Routing code
1

Descriptor code
2
Automation
This message is sent to the system console and to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example

EZY1289I 02/19/09 13:51:39 CICS/SOCKETS LISTENER TRANSACTION CSKM TERMINATING

EZY1291I mm/dd/yy hh:mm:ss LISTENER TRANSACTION transactionid TASKID= taskno ACCEPTING REQUESTS VIA* PORT port

Explanation
The specified transaction can now receive connection requests on the specified port.
This message is issued when any of the following events occur:
• The listener is initialized and was able to connect to its TCP/IP.
• The listener reconnects to its TCP/IP after its TCP/IP has been restarted.
• The listener's socket descriptor table is no longer full and the table is now accepting client connections.

In the message text:

mm/dd/yy
The date (month/day/year) of the message.

hh:mm:ss
The time (hours:minutes:seconds) of the message.

transactionid
The name of the listener's transaction that can now accept new client connections.

taskno
The task number assigned by CICS.

port
The port number on which the listener identified by the transactionid value is listening.

System action
The listener transaction continues.

Operator response
No action needed.

System programmer response
No action needed.

User response
None.

Problem determination
None.

Source
Not applicable.
Module
EZACIC02

Routing code
Not applicable.

Descriptor code
Not applicable.

Example

EZY1291I 01/19/06 10:07:33 LISTENER TRANSACTION= CSKL TASKID= 000079L ACCEPTING REQUESTS VIA PORT 3010

EZY1292E mm/dd/yy hh:mm:ss CANNOT START LISTENER, TRUE NOT ACTIVE TRANSACTION= tran TASKID= cicstask EIBRCODE BYTE3=rr

Explanation
The initialization of the CICS socket interface did not complete successfully and this Listener cannot continue.

System action
Listener transaction tran terminates.

Operator response
If EZAO is being used to start the Listener, ensure that the CICS socket interface has successfully completed initialization first. If this happens during automatic initialization, look for other messages which would indicate why the initialization of the CICS socket interface failed.


System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1293E mm/dd/yy hh:mm:ss INITAPI CALL FAILURE TRANSACTION=tran TASKID= cicstask ERRNO=errno

Explanation
Listener transaction tran experienced a failure on the INITAPI call.
errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
Listener transaction tran terminates.
Operator response
Use the `errno` value to determine the cause of the failure.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1294E mm/dd/yy hh:mm:ss SOCKET CALL FAILURE TRANSACTION= tran TASKID= cicstask ERRNO= `errno`

Explanation
Listener transaction `tran` experienced a failure on the SOCKET call.
`errno` is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
Listener transaction `tran` terminates.

Operator response
Use the `errno` value to determine the cause of the failure.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1295E mm/dd/yy hh:mm:ss BIND CALL FAILURE TRANSACTION= tran TASKID= cicstask ERRNO= `errno`

Explanation
Listener transaction `tran` experienced a failure on the BIND call.
`errno` is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
Listener transaction `tran` terminates.

Operator response
Use the `errno` value to determine the cause of the failure.
Notes:
1. An ERRNO=13 could indicate that the port and jobname specified in the PORT statement in hlq.TCPIP.PROFILE does not match the port and jobname used by the CICS Listener.
2. An ERRNO=48 could indicate that the port is not reserved in hlq.TCPIP.PROFILE.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1296E  mm/dd/yy hh:mm:ss LISTEN CALL FAILURE TRANSACTION= tran TASKID= cicstask ERRNO= errno

Explanation
Listener transaction tran experienced a failure on the LISTEN call.
errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
Listener transaction tran terminates.

Operator response
Use the errno value to determine the cause of the failure.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1297E  mm/dd/yy hh:mm:ss GETCLIENTID CALL FAILURE TRANSACTION= tran TASKID= cicstask ERRNO= errno

Explanation
Listener transaction tran experienced a failure on the GETCLIENTID call.
errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
Listener transaction tran terminates.
Operator response
Use the `errno` value to determine the cause of the failure.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1298E  mm/dd/yy hh:mm:ss CLOSE FAILURE TRANID= tran TASKID= cicstask ERRNO= errno

Explanation
Listener transaction `tran` experienced a failure on the CLOSE call.

`errno` is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
Listener transaction `tran` continues.

Operator response
Use the `errno` value to determine the cause of the failure.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1299E  mm/dd/yy hh:mm:ss SELECT CALL FAILURE TRANSACTION= tran TASKID= xxxxx ERRNO= errno

Explanation
Listener transaction `tran` experienced a failure on the SELECT call.

`errno` is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
Listener transaction `tran` terminates.

Operator response
Use the `errno` value to determine the cause of the failure.
Explanation

The Listener transaction transactionid experienced a failure on the RECV call.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

*transactionid* is the name of the Listener transaction performing the RECV Socket.

*tasknumber* is the CICS task number assigned to the CICS transaction transactionid.

*errno* is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

*inetaddress* is the internet address of the connecting client.

*portnumber* is the connecting client's port number.

System action

The Listener transaction transactionid continues.

Operator response

Use the errno value to determine the cause of the failure.

System programmer response

None.
transactionid is the transaction name of the CICS Listener.

ipaddr is the internet address of the remote client.

port is the port number of the remote client.

**System action**
The Listener transaction transactionid continues.

**Operator response**
Correct the client program.

**System programmer response**
None.

**Module**
EZACIC02

**Procedure name**
LISTENER

**Explanation**
The initial message from the client did not arrive within the read timeout value specified for this Listener in the CICS Sockets configuration file.

mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

inetaddress is the internet address of the connecting client.

portnumber is the connecting client’s port number.

tran_id is the name of the listener’s transaction.

task_id is the task number of the listener’s transaction.

**System action**
The Listener closes the connection socket and does not attempt to start a server transaction.

**Operator response**
Determine the cause of the delay and correct it.

**System programmer response**
None.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: LISTENER

168 z/OS Communications Server: IP Messages: Volume 3 (EZY)
Module
EZACIC02

Routing code
10

Descriptor code
12

Automation
This message is sent to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example
EZY1302I 02/24/09 16:13:16 READ TIMEOUT PARTNER INET ADDR=9.42.105.102 PORT= 1030 LISTENER

EZY1303I  mm/dd/yy hh:mm:ss EZACIC02 GIVESOCKET TIMEOUT TRANS transactionid PARTNER INET ADDR=inetaddress PORT=portnumber

Explanation
The started server transaction did not perform the takesocket within the timeout value specified for this Listener in the CICS Sockets configuration file.

mm/dd/yy is the date (month/day/year) of the message.
hh:mm:ss is the time (hours:minutes:seconds) of the message.
transactionid is the name of the transaction that was requested by the connecting client.
inetaddress is the internet address of the connecting client.
portnumber is the connecting client's port number.

System action
Send an error message to the client and close the socket.

Operator response
Determine the reason for the delay in the server transaction. Possible causes are an overloaded CICS system or excessive processing in the server transaction before the takesocket is issued. Correct the situation and try again.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1306E  mm/dd/yy hh:mm:ss SECURITY EXIT mmmmmmmmm IS NOT DEFINED TRANID= tran TASKID=xxxxxxxxx
Explanation
The security exit specified for this Listener in the CICS Sockets configuration file is not defined to CICS.

System action
Close the socket and terminate the connection.

Operator response
Use CICS RDO to define the security exit.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1307E  mm/dd/yy hh:mm:ss MAXIMUM # OF SOCKETS USED TRANS= tran TASKID= cicstask
ERRNO= errno

Explanation
All of the sockets allocated to Listener transaction xxxx are in use.
errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

System action
The ACCEPT call is delayed until a socket is available.

Operator response
Use the EZAC transaction to increase the number of sockets allocated Listener tran and then stop and restart Listener transaction tran.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1308E  mm/dd/yy hh:mm:ss ACCEPT CALL FAILURE TRANSACTION= tran TASKID= cicstask ERRNO= errno

Explanation
Listener transaction tran experienced a failure on the ACCEPT call.
errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

**System action**
Listener transaction tran terminates.

**Operator response**
Use the errno value to determine the cause of the failure.

**System programmer response**
None.

**Module**
EZACIC02

**Procedure name**
LISTENER

**EZY1309E** mm/dd/yy hh:mm:ss GIVESOCKET FAILURE TRANS transactionid TASKID=tasknumber
ERRNO=errno INET ADDR=inetaddress PORT=portnumber

**Explanation**
The Listener transaction transactionid experienced a failure on the GIVESOCKET call.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

*transactionid* is the name of the transaction that was requested by the connecting client.

*tasknumber* is the CICS task number assigned to the CICS transaction transactionid.

*errno* is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

*inetaddress* is the internet address of the connecting client.

*portnumber* is the connecting client's port number.

**System action**
The Listener transaction transactionid terminates.

**Operator response**
Use the errno value to determine the cause of the failure.

**System programmer response**
None.

**Module**
EZACIC02

**Procedure name**
LISTENER
**EZY1310E**  
*mm/dd/yy hh:mm:ss IC VALUE NOT NUMERIC TRANID=transactionid PARTNER INET ADDR/inetaddress PORT=portnumber*

**Explanation**
The interval specified in the transaction input message contains one or more non-numeric characters.  
*mm/dd/yy* is the date (month/day/year) of the message.  
*hh:mm:ss* is the time (hours:minutes:seconds) of the message.  
*transactionid* is the name of the transaction that was requested by the connecting client.  
*inetaddress* is the internet address of the connecting client.  
*portnumber* is the connecting client's port number.

**System action**
The interval is ignored, and the transaction is started immediately.

**Operator response**
Correct the client program which is sending this transaction input message.

**System programmer response**
None.

**Module**
EZAICIC02

**Procedure name**
LISTENER

---

**EZY1311E**  
*mm/dd/yy hh:mm:ss CICS TRANID transactionid NOT AUTHORIZED PARTNER INET ADDR/inetaddress PORT=portnumber*

**Explanation**
The transaction name specified in the transaction input message is not RSL authorized.  
*mm/dd/yy* is the date (month/day/year) of the message.  
*hh:mm:ss* is the time (hours:minutes:seconds) of the message.  
*transactionid* is the name of the transaction that was requested by the connecting client.  
*inetaddress* is the internet address of the connecting client.  
*portnumber* is the connecting client's port number.

**System action**
The transaction is not started.

**Operator response**
Correct the CICS transaction definition if the transaction should be authorized or the client program if it is sending the wrong transaction name.
System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

**EZY1312E**  
_mm/dd/yy hh:mm:ss SECURITY EXIT mmmmmmmmm CANNOT BE LOADED TRANID= tran  
TASKID=cicstask

Explanation
Listener transaction _tran_ experienced a failure when it attempted to load security exit program _mmmmmmmm_.

System action
Listener transaction _tran_ continues but the server transaction associated with this transaction input message is not started.

Operator response
Use CEMT to determine the status of the exit program and correct whatever problems are found.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

**EZY1313E**  
_mm/dd/yy hh:mm:ss LISTENER NOT AUTHORIZED TO ACCESS SECURITY EXIT  
_mmmmmmmmm TRANID= tran TASKID=xxxxxxx

Explanation
Listener transaction _tran_ is not authorized to access security exit program _mmmmmmmm_.

System action
Listener transaction _tran_ continues but the server transaction associated with this transaction input message is not started.

Operator response
If the security exit program name is incorrect, use EZAC to correct the definition of this Listener on the CICS Sockets Configuration file. If the security exit program is correct, use the CICS RDO facility to authorize Listener transaction _xxxx_ to use security exit program _mmmmmmmm_.

System programmer response
None.
Module
EZACIC02

Procedure name
LISTENER

**EZY1314E**
m/dd/yy hh:mm:ss SECURITY EXIT mmmmmm IS DISABLED
TRANID= tran
TASKID=xxxxxxxx

**Explanation**
Security exit program mmmmmm is disabled.

**System action**
Listener transaction tran continues but the server transaction associated with this transaction input message is not started.

**Operator response**
Use CEMT to enable the security exit program.

**System programmer response**
None.

Module
EZACIC02

Procedure name
LISTENER

**EZY1315E**
m/dd/yy hh:mm:ss INVALID TRANSID transactionid PARTNER INET ADDR= inetaddress
PORT= portnumber

**Explanation**
The transaction input message from the client specified transaction transactionid but this transaction is not defined to CICS.
m/dd/yy is the date (month/day/year) of the message.
hh:mm:ss is the time (hours:minutes:seconds) of the message.
transactionid is the name of the transaction that was requested by the connecting client. The transactionid field will be blank if no printable name was passed by the client or the security exit.
inetaddress is the internet address of the connecting client.
portnumber is the connecting client's port number.

**System action**
The Listener transaction continues but the server transaction associated with this transaction input message is not started.

**Operator response**
If the transaction name is incorrect, correct the client program. If the transaction name is correct, correct the CICS transaction definition.
System programmer response

If transactionid is blank, then there is a possible mismatch because the Listener is expecting the first message segment to start with a transaction name but it does not. A packet trace might be helpful in determining whether there is such a mismatch. For example, if the packet trace shows that the first message segment starts with X'160300' or X'160301' then possibly a clienthello message was received, which indicates that there is an Application Transparent Transport Layer Security (AT-TLS) policy on the client side of the TCP connection but no matching AT-TLS policy (or AT-TLS is not enabled) on the Listener side of the TCP connection. This would need to be addressed by the AT-TLS administrator. See Application Transparent Transport Layer Security Data Protection in z/OS Communications Server: IP Configuration Guide and Steps for diagnosing AT-TLS problems in z/OS Communications Server: IP Diagnosis Guide for more information.

Module

EZACIC02

Procedure name

LISTENER

EZY1316E mm/dd/yy hh:mm:ss TRANSID transactionid IS DISABLED PARTNER INET ADDR=inetaddress PORT=portnumber

Explanation

Transaction transactionid is disabled.

mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

transactionid is the name of the transaction that was requested by the connecting client.

inetaddress is the internet address of the connecting client.

portnumber is the connecting client’s port number.

System action

The Listener transaction continues but the server transaction associated with this transaction input message is not started.

Operator response

Use CEMT to enable the server transaction.

System programmer response

None.

Module

EZACIC02

Procedure name

LISTENER

EZY1317E mm/dd/yy hh:mm:ss TRANSID transactionid IS NOT AUTHORIZED PARTNER INET ADDR=inetaddress PORT=portnumber

EZY1xxx messages 175
Explanation
The Listener transaction `transactionid` is not authorized to start the transaction name specified in the transaction input message.

`mm/dd/yy` is the date (month/day/year) of the message.

`hh:mm:ss` is the time (hours:minutes:seconds) of the message.

`transactionid` is the name of the transaction that was requested by the connecting client.

`inetaddress` is the internet address of the connecting client.

`portnumber` is the connecting client's port number.

System action
The transaction is not started.

Operator response
Authorize Listener transaction `transactionid` to start the transaction.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1318E  `mm/dd/yy hh:mm:ss` TD START SUCCESSFUL QUEUEID=`que`

Explanation
The Listener transaction started a server transaction through transient data queue `que`.

System action
Listener transaction continues and the server transaction is ready to start.

Operator response
None.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1319E  `mm/dd/yy hh:mm:ss` QIDERR FOR TD DESTINATION `queueame` PARTNER INET ADDR=`inetaddress` PORT=`portnumber`
**Explanation**

The Listener transaction was unable to start a CICS transaction through transient data queue *queueName*. DFHRESP was QIDERR.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

*queueName* is the name of the transient data queue that was requested by the connecting client.

*inetAddress* is the internet address of the connecting client.

*portNumber* is the connecting client's port number.

**System action**

The Listener transaction continues.

**Operator response**

If the queue name is incorrect, correct the client program sending this transaction input message. If the queue name is correct, correct the CICS Destination Control Table.

**System programmer response**

None.

**Module**

EZACIC02

**Procedure name**

LISTENER

**EZY1320E**  

**Explanation**

The Listener transaction was unable to start a CICS transaction through transient data queue *queueName*. DFHRESP was IOERR.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

*queueName* is the name of the transient data queue that was requested by the connecting client.

*inetAddress* is the internet address of the connecting client.

*portNumber* is the connecting client's port number.

**System action**

The Listener transaction continues.

**Operator response**

Contact the CICS system programmer.

**System programmer response**

None.
Module
EZACIC02

Procedure name
LISTENER

**EZY1321E**  
_mm/dd/yy hh:mm:ss LENGTH ERROR FOR TD DESTINATION queuename PARTNER INET ADDR=inetaddress PORT=portnumber_

**Explanation**
The Listener transaction was unable to start a CICS transaction through transient data queue *queuename*. DFHRESP was LENGERR.

_mm/dd/yy_ is the date (month/day/year) of the message.

_hh:mm:ss_ is the time (hours:minutes:seconds) of the message.

*queuename* is the name of the transient data queue that was requested by the connecting client.

*inetaddress* is the internet address of the connecting client.

*portnumber* is the connecting client's port number.

**System action**
The Listener transaction continues.

**Operator response**
Contact the CICS system programmer. The minimum length for this queue should be greater than 72.

**System programmer response**
Change definition of Transient Data Queue to accommodate length of this message.

Module
EZACIC02

Procedure name
LISTENER

**EZY1322E**  
_mm/dd/yy hh:mm:ss TD DESTINATION queuename DISABLED PARTNER INET ADDR=inetaddress PORT=portnumber_

**Explanation**
The Listener transaction was unable to start a CICS transaction through transient data queue *queuename*. DFHRESP was DISABLED.

_mm/dd/yy_ is the date (month/day/year) of the message.

_hh:mm:ss_ is the time (hours:minutes:seconds) of the message.

*queuename* is the name of the transient data queue that was requested by the connecting client.

*inetaddress* is the internet address of the connecting client.

*portnumber* is the connecting client's port number.
**System action**
The Listener transaction continues.

**Operator response**
Use CEMT to enable the destination.

**System programmer response**
None.

**Module**
EZACIC02

**Procedure name**
LISTENER

**EZY1323E**  

---

**Explanation**
The Listener transaction was unable to start a CICS transaction through transient data queue `queueiname`. DFHRESP was NOSPACE.

`mm/dd/yy` is the date (month/day/year) of the message.

`hh:mm:ss` is the time (hours:minutes:seconds) of the message.

`queueiname` is the name of the transient data queue that was requested by the connecting client.

`inetaddress` is the internet address of the connecting client.

`portnumber` is the connecting client's port number.

**System action**
The Listener transaction continues.

**Operator response**
Contact the CICS system programmer.

**System programmer response**
Allocate space for this Transient Data Queue.

**Module**
EZACIC02

**Procedure name**
LISTENER

**EZY1324E**  

---

**Explanation**
The Listener transaction was unable to start a CICS transaction through transient data queue `queueiname`.
mm/dd/yyyy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

queue name is the name of the transient data queue that was requested by the connecting client.

inetaddress is the internet address of the connecting client.

portnumber is the connecting client’s port number.

System action
The Listener transaction continues.

Operator response
Contact the CICS system programmer.

System programmer response
Determine the problem with the Transient Data Queue and correct it.

Module
EZACIC02

Procedure name
LISTENER

EZY1325I  mm/dd/yyyy hh:mm:ss START SUCCESSFUL TRANID=transactionid PARTNER INET ADDR/inetaddress PORT=portnumber

Explanation
The Listener transaction was able to start a CICS transaction transactionid transient data queue.

mm/dd/yyyy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

transactionid is the name of the transaction that was requested by the connecting client.

inetaddress is the internet address of the connecting client.

portnumber is the connecting client’s port number.

System action
The Listener transaction continues.

Operator response
None.

System programmer response
None.

Module
EZACIC02
**EZY1326E**  
`mm/dd/yy hh:mm:ss START I/O ERROR TRANID=transactionid PARTNER INET ADDR=inetaddress PORT=portnumber`

**Explanation**
The Listener transaction was unable to start a CICS transaction `transactionid`. DFHRESP was IOERR.  
`mm/dd/yy` is the date (month/day/year) of the message.  
`hh:mm:ss` is the time (hours:minutes:seconds) of the message.  
`transactionid` is the name of the transaction that was requested by the connecting client.  
`inetaddress` is the internet address of the connecting client.  
`portnumber` is the connecting client’s port number.

**System action**
The Listener transaction continues.

**Operator response**
Contact the CICS system programmer.

**System programmer response**
Determine the cause of the I/O error and correct it.

**Module**
EZACIC02

---

**EZY1327E**  
`mm/dd/yy hh:mm:ss START TRANSACTION ID transactionid INVALID PARTNER INET ADDR=inetaddress PORT=portnumber`

**Explanation**
The Listener transaction was unable to start a CICS transaction `transactionid`. DFHRESP was TRANSIDERR.  
`mm/dd/yy` is the date (month/day/year) of the message.  
`hh:mm:ss` is the time (hours:minutes:seconds) of the message.  
`transactionid` is the name of the transaction that was requested by the connecting client.  
`inetaddress` is the internet address of the connecting client.  
`portnumber` is the connecting client’s port number.

**System action**
The Listener transaction continues.

**Operator response**
Contact the CICS system programmer.
System programmer response
Check the transaction definition in RDO to ensure it is correct.

Module
EZACIC02

Procedure name
LISTENER

EZY1328E  

**Explanation**
The Listener transaction was unable to start a CICS transaction `transactionid`. DFHRESP was NOTAUTH. 
- **mm/dd/yy** is the date (month/day/year) of the message.
- **hh:mm:ss** is the time (hours:minutes:seconds) of the message.
- **transactionid** is the name of the transaction that was requested by the connecting client.
- **inetaddress** is the internet address of the connecting client.
- **portnumber** is the connecting client's port number.

System action
The Listener transaction continues.

Operator response
If the transaction ID is incorrect, correct the client program which sent this transaction input message. If the transaction ID is correct, authorize Listener transaction to start this transaction.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1329E  

**Explanation**
The Listener transaction was unable to start a CICS transaction `transactionid`. DFHRESP was 99. 
- **mm/dd/yy** is the date (month/day/year) of the message.
- **hh:mm:ss** is the time (hours:minutes:seconds) of the message.
- **transactionid** is the name of the transaction that was requested by the connecting client.
- **inetaddress** is the internet address of the connecting client.
- **portnumber** is the connecting client's port number.
**System action**
The Listener transaction continues.

**Operator response**
Contact the CICS system programmer.

**System programmer response**
Check the transaction definition in RDO. Look for associated messages in the MSGUSR queue, which might indicate why the transaction would not start.

**Module**
EZACIC02

**Procedure name**
LISTENER

**EZY1330E**  

<table>
<thead>
<tr>
<th>mm/dd/yy</th>
<th>hh:mm:ss</th>
<th>IC START SUCCESSFUL</th>
<th>TRANID=transactionid</th>
<th>PARTNER INET ADDR=inetaddress</th>
<th>PORT=portnumber</th>
</tr>
</thead>
</table>

**Explanation**
The Listener transaction was able to start a CICS transaction `transactionid`.  
`mm/dd/yy` is the date (month/day/year) of the message.  
`hh:mm:ss` is the time (hours:minutes:seconds) of the message.  
`transactionid` is the name of the transaction that was requested by the connecting client.  
`inetaddress` is the internet address of the connecting client.  
`portnumber` is the connecting client's port number.

**System action**
The Listener transaction continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZACIC02

**Procedure name**
LISTENER

**EZY1331E**  

<table>
<thead>
<tr>
<th>mm/dd/yy</th>
<th>hh:mm:ss</th>
<th>IC START I/O ERROR</th>
<th>TRANID=transactionid</th>
<th>PARTNER INET ADDR=inetaddress</th>
<th>PORT=portnumber</th>
</tr>
</thead>
</table>

**Explanation**
Listener transaction was unable to start a CICS transaction `transactionid`. DFHRESP was IOERR.
mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

transactionid is the name of the transaction that was requested by the connecting client.

inetaddress is the internet address of the connecting client.

portnumber is the connecting client's port number.

**System action**
Listener transaction continues.

**Operator response**
Contact the CICS system programmer.

**System programmer response**
Look for other messages in the MSGUSR queue, which provide specific information on the I/O error and correct the problem.

**Module**
EZACIC02

**Procedure name**
LISTENER

EZY1332E mm/dd/yy hh:mm:ss IC START INVALID REQUEST TRANID=transactionid PARTNER INET ADDR=inetaddress PORT=portnumber

**Explanation**
Listener transaction was unable to start a CICS transaction transactionid. DFHRESP was INVREQ.

mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

transactionid is the name of the transaction that was requested by the connecting client.

inetaddress is the internet address of the connecting client.

portnumber is the connecting client's port number.

**System action**
Listener transaction continues.

**Operator response**
Collect the messages written to the console and MSGUSR queue, client input data, and a SOCKAPI component trace and contact the IBM Software Support Center.

**System programmer response**
None.

**Module**
EZACIC02
**Listener Transaction Start Failed**

**Explanation**

Listener transaction was unable to start a CICS transaction `transactionid`.  
`mm/dd/yy` is the date (month/day/year) of the message.  
`hh:mm:ss` is the time (hours:minutes:seconds) of the message.  
`transactionid` is the name of the transaction that was requested by the connecting client.  
`inetaddress` is the internet address of the connecting client.  
`portnumber` is the connecting client's port number.  

**System action**

Listener transaction continues.  

**Operator response**

Contact the CICS system programmer.  

**System programmer response**

Check the RDO definition of the transaction. Collect the messages written to the console and MSGUSR queue, client input data, and a SOCKAPI component trace and contact the IBM Software Support Center.  

**Module**

EZACIC02  

**Listener Transaction Failed**

**Explanation**

This message indicates that the user security exit has given the Listener an invalid USERID field.  
`mm/dd/yy` is the date (month/day/year) of the message.  
`hh:mm:ss` is the time (hours:minutes:seconds) of the message.  
`transactionid` is the name of the transaction that was requested by the connecting client.  
`inetaddress` is the internet address of the connecting client.  
`portnumber` is the connecting client's port number.  
`userid` is the user ID assigned by the user security exit.  

**System action**

The server transaction that is identified by the `transactionid` value does not start.
Operator response
Correct the user ID that is not valid in the user security exit.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1335E mm/dd/yy hh:mm:ss WRITE FAILED ERRNO=errno TRANID=transactionid. PARTNER INET ADDR=inetaddress PORT=portnumber

Explanation
Listener transaction had a failure on a WRITE command.

mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

transactionid is the name of the transaction that was requested by the connecting client.

inetaddress is the internet address of the connecting client.

portnumber is the connecting client's port number.

System action
The Listener transaction continues.

Operator response
Use the errno value to determine the cause of the failure.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1336E mm/dd/yy hh:mm:ss TAKESOCKET FAILURE TRANS transactionid TASKID=tasknumber ERRNO=errno INET ADDR=inetaddress PORT=portnumber

Explanation
The Listener transaction had a failure on a TAKESOCKET command.

mm/dd/yy is the date (month/day/year) of the message.
hh:mm:ss is the time (hours:minutes:seconds) of the message.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

transactionid is the name of the transaction that was requested by the connecting client.

inetaddress is the internet address of the connecting client.

portnumber is the connecting client's port number.

System action
The Listener transaction continues.

Operator response
Use the errno value to determine the cause of the failure.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1337E  mm/dd/yy hh:mm:ss CICS IN QUIESCE, LISTENER TERMINATING TRANSID= tran TASKID= cicstask

Explanation
Listener transaction tran is terminating because it detected a CICS quiesce in progress.

System action
Listener transaction tran terminates.

Operator response
None.

System programmer response
None.

Module
EZACIC02

Procedure name
LISTENER

EZY1338E  mm/dd/yy hh:mm:ss PROGRAM programname NOT FOUND TRANSID=transactionid PARTNER INET ADDR=inetaddress PORT=portnumber
Explanation
The Listener checked the status of the program associated with the transaction. It was not found.

\textit{mm/dd/yy} is the date (month/day/year) of the message.

\textit{hh:mm:ss} is the time (hours:minutes:seconds) of the message.

\textit{programname} is the name of the program which is associated with the transaction requested by the connecting client.

\textit{transactionid} is the name of the transaction that was requested by the connecting client.

\textit{inetaddress} is the internet address of the connecting client.

\textit{portnumber} is the connecting client's port number.

System action
Listener continues.

Operator response
If \textit{transactionid} is incorrect, correct the client program that sent the transaction input message. If the transaction ID is correct, check the transaction and program definitions in CICS.

System programmer response
None.

Module
EZACIC02

\begin{verbatim}
EZY1339E  mm/dd/yy hh:mm:ss EXIT PROGRAM (EZACIC01) IS NOT ENABLED. DISABLE IGNORED
TERM=term TRAN=tranxxx
\end{verbatim}

Explanation
A termination of the CICS socket interface was requested but the interface is not enabled.

System action
The termination request is ignored.

Operator response
None.

System programmer response
None.

Module
EZACIC22

Procedure name
TERMINATION

\begin{verbatim}
EZY1340E  mm/dd/yy hh:mm:ss API ALREADY QUIESCING DUE TO PREVIOUS REQ. EZAO IGNORED
TERM=term TRAN=tranxxx
\end{verbatim}
Explanation
A request for a quiesce of the CICS socket interface has been made but one is already in progress.

System action
Ignore the second request.

Operator response
None.

System programmer response
None.

Module
EZACIC22

Procedure name
TERMINATION

EZY1341E  mm/dd/yy hh:mm:ss API ALREADY IN IMMED MODE DUE TO PREV. REQ. EZAO IGNORED
TERM=term TRAN=tranxxx

Explanation
A request for an immediate of the CICS socket interface has been made but one is already in progress.

System action
Ignore the second request.

Operator response
None.

System programmer response
None.

Module
EZACIC22

Procedure name
TERMINATION

EZY1342I  mm/dd/yy hh:mm:ss DISABLE DELAYED UNTIL ALL USING TASKS COMPLETE TERM=termid
TRAN=transid

Explanation
A quiesce is in progress and is waiting for all outstanding CICS tasks to complete using the CICS socket interface. When an IP CICS interface is being shut down the following actions occur:

• All listeners are posted to end.
• If the interface is configured as OTE=NO, then all non-listener tasks have their MVS subtask posted and their CICS task ends.
• If the interface is configured as OTE=YES, then any non-listener transaction that is running a blocking socket command is forced to end by a CICS FORCE PURGE action.

See the information about the TYPE=CICS parameter in z/OS Communications Server: IP CICS Sockets Guide for information about the OTE configuration option.

In the message text:

\textit{mm/dd/yy} \\
\textbullet\textit{hh:mm:ss} \\
\textit{termid} \\
\textit{transid}

\textbf{System action}

The system continues to shut down.

\textbf{Operator response}

None.

\textbf{System programmer response}

None.

\textbf{Module}

EZACIC22

\textbf{Procedure name}

TERMINATION

\textbf{EZY1343I} \textit{mm/dd/yy hh:mm:ss} CICS/SOCKETS INTERFACE IMMEDIATELY DISABLED TERM=\textit{term} TRAN=\textit{tranxxx}

\textbf{Explanation}

A request for the immediate ending of the CICS socket interface has been successfully completed.

In the message text:

\textit{mm/dd/yy} \\
\textbullet\textit{hh:mm:ss} \\
\textit{term} \\
\textit{tran}

\textbf{System action}

The CICS socket interface ends.
Operator response
None.

System programmer response
None.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CICS socket interface termination

Module
EZACIC22

Routing code
1

Descriptor code
2

Automation
This message is sent to the system console and to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example
EZY1343I 02/19/09 13:52:50 CICS/SOCKETS INTERFACE IMMEDIATELY DISABLED. TERM= TRA= EZAP

EZY1344I mm/dd/yy hh:mm:ss CICS/SOCKETS INTERFACE QUIESCENTLY DISABLED TERM=term TR=tran

Explanation
A request for the deferred ending of the CICS socket interface has been successfully completed.

In the message text:

\textit{mm/dd/yy} 
\hfill The date (month/day/year) of the message.

\textit{hh:mm:ss} 
\hfill The time (hours:minutes:seconds) of the message.

\textit{term} 
\hfill The terminal ID from which the command to end the CICS socket interface was issued.

\textit{tran} 
\hfill The transaction ID that is ending the CICS socket interface.

System action
The CICS socket interface ends.
Operator response
None.

System programmer response
None.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CICS socket interface termination

Module
EZACIC22

Routing code
1

Descriptor code
2

Automation
This message is sent to the system console and to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example
EZY1344I 02/19/09 13:52:21 CICS/SOCKETS INTERFACE QUIESCENTLY DISABLED. TERM= TRAN=EZAP

EZY1347I  mm/dd/yy hh:mm:ss PROGRAM programname ASSUMED TO BE AUTOINSTALLED
TRANID=transactionid IP ADDR=inetaddress PORT=portnumber

Explanation
The Listener checked the status of the program associated with the transaction. It was not found. Because program autoinstall is active in the CICS region, the Listener assumes that the program definition will automatically be installed by CICS.

mm/dd/yy
The date (month/day/year) of the message.

hh:mm:ss
The time (hours:minutes:seconds) of the message.

programname
The name of the undefined program which is associated with the transaction requested by the connecting client.

transactionid
The name of the transaction that was requested by the connecting client.

inetaddress
The internet address of the connecting client.

portnumber
The connecting client's port number.
System action
Listener continues.

Operator response
None.

System programmer response
Verify that the program name in the transaction definition is correct. Verify that the program is intended to be autoinstalled rather than explicitly defined in the PPT.

Module
EZACIC02

Procedure name
LISTENER

EZY1348E  mm/dd/yy hh:mm:ss INVALID SOCKET FUNCTION function ERRNO errno TRAN tranid TASK taskid

Explanation
The task related user exit (TRUE) detected an invalid socket function on a call request from the CICS application program.

mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

function is the invalid socket function.

errno is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERRNOs) in z/OS Communications Server: IP and SNA Codes.

tranid is the name of the CICS transaction.

taskid is the CICS task ID number.

System action
The TRUE is disabled and the task abends with an AEY9 CICS abend code.

Operator response
Correct the invalid socket function and try again.

The most probable errno is 10011 "INVALID SOCKET FUNCTION". If the socket function name appears correct, ensure that the application padded the function call with blanks.

System programmer response
None.

Module
EZACIC01

Procedure name
Task Related User Exit (TRUE)
Explanation
The CICS Listener received an abnormal response from CICS when attempting to open the CICS Sockets configuration file (EZACONFG) using an EXEC CICS SET FILE call.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

*transactionid* is the name of the transaction under which the Listener is executing.

*eibresp2* is the EIBRESP2 value returned by CICS on the EXEC CICS SET FILE call as described at http://www.ibm.com/software/htp/cics/library/.

System action
The Listener ends.

Operator response
Contact the CICS system programmer.

System programmer response
Use the CICS System Programming Reference at http://www.ibm.com/software/htp/cics/library/ to interpret the value of EIBRESP2. If the file is not known to CICS, perform the installation steps for the configuration file.

Module
EZACIC02

Procedure name
LISTENER

Explanation
The IP CICS socket interface uses a CICS EXTRACT EXIT command to determine whether the IP CICS Sockets Task Related User Exit (TRUE) is enabled. This action is performed by IP CICS socket interface initialization and shutdown programs, the Listener, and by any user application linking to the IP CICS domain name server module.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

*api_function* is the CICS command performed.

*action* is the action intended.

- **ENABLE** means the IP CICS socket interface is being enabled.
- **DISABLE** means the IP CICS socket interface is being disabled.
- **STARTUP** means the IP CICS socket interface is being started.

*termid* is the terminal ID where the transaction receiving the error is executing.

*tranid* is the name of the transaction that is incurring the security violation.
System action

- If the TRUE is being enabled when the IP CICS socket interface is initializing, then the enable action is ignored and the interface is not activated.
- If the TRUE is being disabled when the IP CICS socket interface is shutting down, then the disable action is ignored and the interface remains active.
- If the IP CICS socket interface is being started, then the startup action is ignored and the interface remains inactive.

Operator response

Contact the CICS system programmer.

System programmer response

Ensure that the user ID being used is allowed at least UPDATE access to the EXITPROGRAM resource.

Module

EZACIC02, EZACIC21, EZACIC22

Procedure name

Listener, Initialization, Shutdown

**EZY1351E**  
**mm/dd/yy hh:mm:ss**  
EXIT PROGRAM (EZACIC01) IS NOT ENABLED, **action** IGNORED.  
TERM=termid TRAN=transid

Explanation

The IP CICS socket interface uses a CICS ENABLE PROGRAM command to enable the IP CICS Sockets Task Related User Exit (TRUE). This action is performed by IP CICS socket interface initialization.

- **mm/dd/yy** is the date (month/day/year) of the message.
- **hh:mm:ss** is the time (hours:minutes:seconds) of the message.
- **action** is the action intended.
- **ENABLE** means the IP CICS socket interface is being enabled.
- **DISABLE** means the IP CICS socket interface is being disabled.
- **termid** is the terminal ID where the transaction receiving the error is executing.
- **transid** is the name of the transaction that is incurring the security violation.

System action

The IP CICS socket interface is not initialized.

Operator response

Contact the CICS system programmer.

System programmer response

Ensure that the user ID being used is allowed at least UPDATE access to the EXITPROGRAM resource.

Module

EZACIC21
**Procedure name**
Initialization

**EZY1352E mm/dd/yy hh:mm:ss SUBTASK ENDED UNEXPECTEDLY TRANSACTION= transactionid TASKID= taskid**

**Explanation**
The current tasks CICS Sockets subtask ended unexpectedly. This is probably caused by an ABEND of the subtask.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

*transactionid* is the name of the CICS transaction whose subtask ended unexpectedly.

*taskid* is the CICS task number of the task whose subtask ended unexpectedly.

**System action**
The CICS socket interface is disabled for the current task. Any subsequent CICS Sockets calls by that task will result in CICS ABEND code AEY9. Other tasks are not affected.

**Operator response**
Contact the CICS system programmer.

**System programmer response**
Check the console log for previous messages that explain what happened to the subtask.

**Module**
EZACIC01

**Procedure name**
TASK RELATED USER EXIT (TRUE)

**EZY1353E mm/dd/yy hh:mm:ss COMMA MISSING AFTER IC TRANS ID = transactionid PARTNER IP ADDR = inetaddress PORT = portnumber**

**Explanation**
The listener did not find a comma delimiter after the interval control (IC) start type indicator in the client’s transaction request message.

In the message text:

*mm/dd/yy*  
The date (month/day/year) of the message.

*hh:mm:ss*  
The time (hours:minutes:seconds) of the message.

*transactionid*  
The name of the transaction that was requested by the connecting client.

*inetaddress*  
The internet address of the connecting client.

*portnumber*  
The connecting client’s port number.
System action
The listener does not start the transaction specified by the client's transaction request message and ends the connection. This message is also returned to the client.

Operator response
Ensure that a comma delimiter separates the IC start type and the IC start time. See Listener input format in z/OS Communications Server: IP CICS Sockets Guide for information about the client's transaction request message.

System programmer response
None.

User response
Not applicable.

Problem determination
Not applicable.

Source
Module
EZACIC02

Routing code
Not applicable.

Descriptor code
Not applicable.

Example
An example of a transaction request message for the standard listener:

```
SCCS,DATA,IC000010
EZY1258I 10/11/05 14:01:55 EZACIC02 ENTRY POINT IS 17CB2028
EZY1258I 10/11/05 14:01:55 EZACIC01 ENTRY POINT IS 177E2518
EZY1291I 10/11/05 14:01:56 LISTENER TRANSACTION= CSKL TASKID= 0000032L ACCEPTING REQUESTS VIA PORT 3010
EZY1353E 10/11/05 14:02:56 COMMA MISSING AFTER IC TRANSACTION ID= SCCS PARTNER INET ADDR=10.1.1.2 PORT= 1076
```

**EZY1354I**  
*mm/dd/yy hh:mm:ss CICS/ SOCKETS CICS TRACING IS status*

Explanation
This message shows the status of changing IP CICS Sockets CICS tracing and is issued when one of the following occurs:

- The operator issued the EZAO,START,TRACE transaction.
- The operator issued the EZAO,STOP,TRACE transaction.
- The CICS Master User Trace Flag is specified as OFF and the IP CICS Sockets TRACE configuration is specified as YES.
**status** is the status of CICS tracing for the IP CICS socket interface.

- **ENABLED** indicates that the IP CICS socket interface will generate CICS trace data when CICS tracing is active.
- **DISABLED** indicates that the IP CICS socket interface will not generate CICS trace data.

**System action**

When **status** is **ENABLED**, IP CICS Sockets will generate CICS trace data when CICS tracing is active. When **status** is **DISABLED**, IP CICS Sockets will not generate CICS trace data.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZACIC00, EZACIC01

**Procedure name**
TRC00000, SUB05100

---

**EZY1355I  mm/dd/yy hh:mm:ss CICS/ SOCKETS TCBLIM EXCEEDS MAXOPENTCBS**

**Explanation**
IP CICS Sockets has determined that the value specified for TCBLIM exceeds the value of MAXOPENTCBS allowed at the time the interface was enabled. TCBLIM will be forced to the same value as MAXOPENTCBS.

**System action**
IP CICS Sockets TCBLIM will default to the value of MAXOPENTCBS. IP CICS Sockets processing continues.

**Operator response**
Contact the CICS system programmer.

**System programmer response**
Adjust the value specified by the TCBLIM configuration option using one or more of the following methods:

- Specify an appropriate TCBLIM value on the EZACICD TYPE=CICS,TCBLIM= macro.
- Specify an appropriate TCBLIM value using the EZAC Configuration transaction.
- Specify an appropriate TCBLIM value dynamically by using the EZAO Operator transaction.
- Specify an appropriate MAXOPENTCBS value using the CICS System Initialization parameters.
- Specify an appropriate MAXOPENTCBS value using the CICS Master Terminal transaction, CEMT SET DISPATCHER MAXOPENTCBS.

See the following sections in z/OS Communications Server: IP CICS Sockets Guide:

- Building the configuration data set with EZACICD for information about using the EZACICD macro.
• Configuration transaction (EZAC) for information about the EZAC Configuration transaction.
• SET function and INQUIRE function for information about the EZAO Operator transaction.
• the EZACICD TYPE parameter in z/OS Communications Server: IP CICS Sockets Guide for a description of the TCBLIM parameter.


Module
EZACIC21

Procedure name
Initialization

EZY1356E  mm/dd/yy hh:mm:ss CICS/ SOCKETS TCBLIM HAS BEEN REACHED

Explanation
The number of IP CICS Sockets-enabled CICS tasks using an Open API, L8, TCB is equal to the value specified by the TCBLIM configuration option.

mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

System action
The IP CICS socket interface will suspend any new tasks until one of the following actions occur:
• The IP CICS Sockets TCBLIM value is increased.
• Existing transactions using IP CICS Sockets end.

This message will be issued only when the interface detects that it has reached TCBLIM. EZY1360I will be issued when this condition is relieved.

Operator response
Contact the CICS system programmer.

System programmer response
Use the CICS Master Terminal transaction, CEMT INQ TASK HVALUE(ATTCBLIM), to determine which IP CICS Sockets-enabled CICS transactions are subject to TCBLIM. Either take action to reduce the IP CICS Sockets work load or increase the IP CICS Socket TCBLIM configuration option. You can use the EZAO,SET,CICS Operator transaction to dynamically increase TCBLIM. The new value you set for the TCBLIM configuration option must be less than or equal to the value specified by MAXOPENTCBS.

Module
EZACIC01

Procedure name
SUB16000

EZY1357I  mm/dd/yy hh:mm:ss TRANSIENT DATA QUEUE SPECIFIED ON ERRORTD IS NOT DEFINED TO CICS

EZY1xxx messages 199
**Explanation**  
IP CICS Sockets has determined that the CICS transient data queue specified by the ERRORTD configuration option was not defined to the CICS region where the IP CICS socket interface is enabled.  

*mm/dd/yy* is the date (month/day/year) of the message.  

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

**System action**  
The CSMT transient data queue will be used for reporting all IP CICS Sockets interface messages. CSMT is the default CICS transient data queue name.

**Operator response**  
Contact the CICS system programmer.

**System programmer response**  
Ensure that the CICS transient data queue specified by the ERRORTD configuration option is properly defined to CICS.  
See the Transient data definition in z/OS Communications Server: IP CICS Sockets Guide for more information.

**Module**  
EZACIC21

**Procedure name**  
Initialization

---

**EZY1358E  10999 ABEND - IP CICS SOCKETS USING OTE**

**Explanation**  
IP CICS Sockets has incorrectly called the MVS subtask wrapper module when the interface was enabled to use CICS Open Transaction Environment.

**System action**  
The IP CICS socket interface will stop.

**Operator response**  
Contact the CICS system programmer.

**System programmer response**  

**Module**  
EZACIC03

**Procedure name**  
MVS SUBTASK

---

**EZY1359I  mm/dd/yy hh:mm:ss CICS/SOCKETS APPLICATIONS WILL USE THE QR TCB**
**Explanation**

IP CICS Sockets has determined that CICS FORCEQR=YES is specified.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

**System action**

CICS will force all user application programs, including those enabled to IP CICS Sockets, that are specified as threadsafe to run under the CICS Quasi-Reentrant (QR) TCB, as if they were specified as quasi-reentrant programs.

**Operator response**

Contact the CICS system programmer.

**System programmer response**

If you do not want to incur the overhead of CICS switching Open API-enabled tasks back to the QR TCB, then change the value of FORCEQR to NO. See [http://www.ibm.com/software/htp/cics/library/](http://www.ibm.com/software/htp/cics/library/) for more information about the FORCEQR CICS System Initialization parameter. See [http://www.ibm.com/software/htp/cics/library/](http://www.ibm.com/software/htp/cics/library/) for more information about the CICS Master Terminal transaction that is used to dynamically change the FORCEQR setting.

**Module**

EZACIC21

**Procedure name**

Initialization

---

**EZY1360I  mm/dd/yy hh:mm:ss CICS/ SOCKETS TCBLIM CONDITION HAS BEEN RELIEVED**

**Explanation**

IP CICS Sockets enable transactions are no longer suspended due to TCBLIM.

*mm/dd/yy* is the date (month/day/year) of the message.

*hh:mm:ss* is the time (hours:minutes:seconds) of the message.

**System action**

Any new or suspended IP CICS Sockets work will now be processed without being suspended due to IP CICS Sockets being at TCBLIM.

**Operator response**

None.

**System programmer response**

None.

**Module**

EZACIC01
**Explanation**
The IP CICS Sockets OTE configuration parameter is specified as YES. IP CICS Sockets determined that the CICS environment that is required to support the exploitation of CICS Open Transaction Environment by IP CICS Sockets is not available.

`mm/dd/yy` is the date (month/day/year) of the message.

`hh:mm:ss` is the time (hours:minutes:seconds) of the message.

**System action**
The IP CICS socket interface is not enabled to use CICS Open Transaction Environment.

**Operator response**
Contact the system programmer.

**System programmer response**
Perform one of the following:

- Upgrade the level of CICS to support Open Transaction Environment. The CICS Open Transaction Environment requires CICS/TS V2R2 or later.

- Change the IP CICS socket interface configuration to use MVS subtasks when configuring it by using the EZAC configuration transaction or the EZACICD macro.

**Module**
EZACIC21

**Procedure name**
Initialization

**Explanation**
CICS Sockets attempted to start the specified listener, but the EXEC CICS START command failed with the RESP1 and RESP2 values listed in the message text.

`mm/dd/yy` is the date (month/day/year) of the message.

`hh:mm:ss` is the time (hours:minutes:seconds) of the message.

`transactionid` is the transaction name of the listener that the CICS Sockets attempted to start.

`resp1` is the RESP1 value returned by the EXEC CICS START transaction.

`resp2` is the RESP2 value returned by the EXEC CICS START transaction.

**System action**
The CICS Listener does not start.
Operator response
None.

System programmer response

- If the RESP2 value is 8 or 9, then the problem is related to the USERID parameter in the definition of the listener. Verify that the USERID parameter is correct. See Setting up and configuring CICS TCP/IP in z/OS Communications Server: IP CICS Sockets Guide for a description of the USERID parameter.
- If the RESP2 value is 8, then the USERID parameter of the listener definition specifies a user ID that is not known to RACF®. Therefore, either change the USERID parameter or define the user ID to RACF.
- If the RESP2 value is 9, then the user ID under which the EXEC CICS START was issued does not have SURROGAT security access to the user ID that is specified in the USERID parameter. For example, if the failure occurs during CICS PLT processing, then the PLT user ID does not have SURROGAT security access to the listener's user ID. See http://www.ibm.com/software/htp/cics/library/ for more information.

Module
EZACIC21

Procedure name
INITIALIZATION

EZY1363I    mm/dd/yy hh:mm:ss LISTENER transactionid taskno HAD threads THREADS ACTIVE WHEN STACK tcpname ENDED

Explanation
This message displays the number of listener threads that were active when the TCP/IP stack that is specified ended. This message is followed by one or more EZY1368I messages that describe the clients that are affected. In the message text:

mm/dd/yy
  The date (month/day/year) of the message.

hh:mm:ss
  The time (hours:minutes:seconds) of the message.

transactionid
  The listener’s transaction ID.

taskno
  The task number assigned by CICS.

threads
  The number of threads that were active when the specified TCP/IP stack ended.

tcpname
  The TCP/IP procedure name with which the listener had affinity.

System action
Processing continues.

Operator response
No action needed.
System programmer response
No action needed.

User response
No action needed.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CICS Socket Interface and API

Module
EZACIC02

Routing code
10

Descriptor code
12

Automation
This message is sent to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example
Following is an example of the messages that are displayed when the stack has ended while the listener was processing data.

EZY1369E 01/10/06 12:59:32 LISTENER CSKL 10295 IS DELAYED, STACK TCPCS IS UNAVAILABLE
EZY1363I 01/10/06 12:59:33 LISTENER CSKL 10295 HAD 5 THREADS ACTIVE WHEN STACK TCPCS ENDED
EZY1367I 01/10/06 12:59:33 SOCKET IP ADDRESS
EZY1368I 01/10/06 12:59:33 2 10.11.1.2 10245 PAYR
EZY1368I 01/10/06 12:59:33 12 2001:DB8:10::11:2:1 21089
EZY1368I 01/10/06 12:59:33 15 10.81.1.1 10245 PAYR
EZY1368I 01/10/06 12:59:33 19 10.81.1.1 21212 ACCT
EZY1368I 01/10/06 12:59:33 999 2001:DB8:10::11:1:2 00901 ORDR

EZY1364I mm/dd/yy hh:mm:ss LISTENER transactionid DETECTED THAT TTLS IS status ON STACK tcpname

Explanation
The CICS Listener is defined with a GETTID parameter of YES which indicates that the listener is requested to attempt to obtain the connecting client certificates and user IDs from Application Transparent Transport Layer Security (AT-TLS). If status is DISABLED, then AT-TLS is disabled in the TCP/IP stack. Therefore, the listener is unable to obtain client certificates and user IDs as requested by the GETTID parameter. If status is ENABLED, then AT-TLS has been enabled in the TCP/IP stack, making it possible for the listener to obtain client certificates and user IDs.

mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

transactionid is the name of the listeners CICS transaction.
**status** is the status of AT-TLS in the TCP/IP stack. **status** is either **DISABLED** or **ENABLED**.

**tcpname** is the name of the TCP/IP stack.

**System action**
The listener continues its normal processing, which includes attempting to obtain client certificates and User IDs.

**Operator response**
Contact the system programmer.

**System programmer response**
No response is needed if **status** is **ENABLED**. If **status** is **DISABLED**, then verify that the GETTID parameter of YES is correct in the listener definition. If so, request that your AT-TLS administrator investigate why AT-TLS is not enabled in the TCP/IP stack. See Setting up and configuring CICS TCP/IP in z/OS Communications Server: IP CICS Sockets Guide for a description of the GETTID parameter.


**Module**
EZACIC02

**Procedure name**
LISTENER

**EZY1365E**  

**mm/dd/yy hh:mm:ss** LISTENER **transactionid** **taskno** IS NOT ACCEPTING REQUESTS ON PORT **port**

**Explanation**
The listener identified by the specified transaction ID and task number cannot process inbound connections because the listener's socket descriptor table is full.

In the message text:

**mm/dd/yy**

The date (month/day/year) of the message.

**hh:mm:ss**

The time (hours:minutes:seconds) of the message.

**transactionid**

The name of the listener's transaction that cannot accept new client connections.

**taskno**

The task number assigned by CICS.

**port**

The port number on which the specified listener is listening.

**System action**
The listener does not accept new connections until the number of socket descriptors currently being processed by the listener is less than the value specified by the lesser of either the system MAXFILEPROC parameter or the listener user ID's FILEPROCMAX parameter.
Operator response
Contact the system programmer.

System programmer response
Perform any of the following actions as appropriate:

- If the ERRORTD log indicates that the child server transaction failed to take the client's given socket, then investigate the CICS region where the child server transaction runs.

  See the steps for diagnosing TCP/IP clients that are unable to connect in z/OS Communications Server: IP Diagnosis Guide for information about diagnosing child server transactions problems.


- If the listeners NUMSOCK value is greater than or equal to the value specified by the MAXFILEPROC parameter, then perform one of the following actions:
  - Set the NUMSOCK value to be less than the MAXFILEPROC value using either the EZACICD macro or the EZAC configuration transaction and then restart the listener. See the information about configuring a CICS TCP/IP environment in z/OS Communications Server: IP CICS Sockets Guide for more information about using the EZACICD macro and the EZAC configuration transaction.
  - Set the MAXFILEPROC value to be greater than the NUMSOCK value using the SETOMVS system command. See the SETOMVS command information in z/OS MVS System Commands for information about dynamically changing the MAXFILEPROC option that z/OS UNIX System Services is currently using.

- If the listener user ID FILEPROCMAX value is less than the value specified by the NUMSOCK parameter, set the FILEPROCMAX value to be greater than the value specified by the NUMSOCK parameter. For more information about the FILEPROCMAX specification, see the documentation provided for the SAF product that is in use on your system. If you are using RACF, see the information about the FILEPROCMAX parameter in the z/OS Security Server RACF Security Administrator's Guide.

User response
No action needed.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CICS Socket Interface and API

Module
EZACIC02

Routing code
1

Descriptor code
2

Automation
This message is sent to the system console and to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.
Example

EZY1365E 01/19/06 10:07:33 LISTENER CSKL 0000079 IS NOT ACCEPTING REQUESTS AT PORT 3010

EZY1366E mm/dd/yy hh:mm:ss CICS/ SOCKETS LISTENER TRANSACTION tranid IS ALREADY ACTIVE

Explanation
The IP CICS Sockets Listener determined that another listener with the same transaction ID is already active.

mm/dd/yy is the date (month/day/year) of the message.

hh:mm:ss is the time (hours:minutes:seconds) of the message.

tranid is the CICS transaction identifier of the duplicate IP CICS Sockets Listener.

System action
The IP CICS Sockets Listener that issued this message ends.

Operator response
Contact the system programmer.

System programmer response
Change the Listeners CICS transaction identifier or port number to ensure that the definition is unique. See Setting up and configuring CICS TCP/IP in z/OS Communications Server: IP CICS Sockets Guide for more information about configuring the IP CICS Sockets Listener.

Module
EZACIC02

Procedure name
Initialization

EZY1367I mm/dd/yy hh:mm:ss SOCK# IP ADDRESS PORT CHILD

Explanation
The listener was processing client connections when its TCP/IP stack ended. This message is issued when the listener has accepted sockets that were not taken by child server tasks. This message is a header message for the EZY1368I detail messages that follow. This message accompanies an EZY1363I message.

In the message text:

mm/dd/yy
The date (month/day/year) of the message.

hh:mm:ss
The time (hours:minutes:seconds) of the message.

System action
Processing continues.

Operator response
No action needed.
System programmer response
No action needed.

User response
No action needed.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CICS Socket Interface and API

Module
EZACIC02

Routing code
10

Descriptor code
12

Automation
This message is sent to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example
Following is an example of the messages displayed when the stack has ended while the listener was processing data.

<table>
<thead>
<tr>
<th>mm/dd/yy</th>
<th>hh:mm:ss</th>
<th>sock#</th>
<th>ipaddr</th>
<th>port</th>
<th>tran</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/10/06</td>
<td>12:59:32</td>
<td>1</td>
<td>10.11.1.2</td>
<td>10245</td>
<td>PAYR</td>
</tr>
<tr>
<td>01/10/06</td>
<td>12:59:33</td>
<td>12</td>
<td>2001:DB8:10::11:2:1</td>
<td>21089</td>
<td></td>
</tr>
<tr>
<td>01/10/06</td>
<td>12:59:33</td>
<td>15</td>
<td>10.91.1.1</td>
<td>10245</td>
<td>INVN</td>
</tr>
<tr>
<td>01/10/06</td>
<td>12:59:33</td>
<td>19</td>
<td>10.81.1.1</td>
<td>21212</td>
<td>ACCT</td>
</tr>
<tr>
<td>01/10/06</td>
<td>12:59:33</td>
<td>999</td>
<td>2001:DB8:10::11:1:2</td>
<td>08901</td>
<td>ORDR</td>
</tr>
</tbody>
</table>

EZY1368I

Explanation
The listener was processing client connections when its TCP/IP stack ended. This message is issued when the listener has accepted sockets that were not taken by child server tasks. One EZY1368I message is issued for each client connection that is being processed.

In the message text:

- **mm/dd/yy**
  The date (month/day/year) of the message.

- **hh:mm:ss**
  The time (hours:minutes:seconds) of the message.

- **sock#**
  The listener’s socket number.
ipaddr
The client's IP address.

port
The client's port number.

tran
The child server's transaction ID. A blank child server transaction ID indicates that the ID has not yet been determined.

System action
Processing continues.

Operator response
No action needed.

System programmer response
No action needed.

User response
No action needed.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CICS Socket Interface and API

Module
EZACIC02

Routing code
10

Descriptor code
12

Automation
This message is sent to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example
Following is an example of the messages displayed when the stack has ended while the listener was processing data.

<table>
<thead>
<tr>
<th>Time</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>12:59:32</td>
<td>EZY1369E 01/10/06 LISTENER CSKL 10295 IS DELAYED, STACK TCPCS IS UNAVAILABLE</td>
</tr>
<tr>
<td>12:59:33</td>
<td>EZY1363I 01/10/06 LISTENER CSKL 10295 HAD 5 THREADS ACTIVE WHEN STACK TCPCS ENDED</td>
</tr>
<tr>
<td>12:59:33</td>
<td>EZY1367I 01/10/06 SOCK# IP ADDRESS PORT CHILD 2 10.11.1.2 10245 PAYR</td>
</tr>
<tr>
<td>12:59:33</td>
<td>EZY1361I 01/10/06 12:59:33 12 2001:DB8:10::11:2:1 21089</td>
</tr>
<tr>
<td>12:59:33</td>
<td>EZY1361I 01/10/06 12:59:33 15 10.91.1.1 10245 INVN</td>
</tr>
<tr>
<td>12:59:33</td>
<td>EZY1361I 01/10/06 12:59:33 19 10.81.1.1 21212 ACCT</td>
</tr>
<tr>
<td>12:59:33</td>
<td>EZY1361I 01/10/06 12:59:33 999 2001:DB8:10::11:1:2 00901 ORDR</td>
</tr>
</tbody>
</table>
EZY1369E  mm/dd/yy hh:mm:ss LISTENER transactionid taskno IS DELAYED, STACK tcpname IS UNAVAILABLE.

Explanation
The TCP/IP stack assigned to the specified listener is not active.
In the message text:

mm/dd/yy
The date (month/day/year) of the message.

hh:mm:ss
The time (hours:minutes:seconds) of the message.

transactionid
The listener's transaction ID.

taskno
The task number assigned by CICS.

tcpname
The TCP/IP procedure name with which the listener had affinity.

System action
The listener releases any resources and connects to the TCP/IP stack specified by the tcpname value. If the connection fails because the stack is not active, then the listener delays using the time value specified by its RTYTIME configuration option and attempts to reconnect. See the TYPE=LISTENER information in z/OS Communications Server: IP CICS Sockets Guide for information about setting the listener's RTYTIME value.

Operator response
Start or restart the TCP/IP address space specified by the tcpname value.

System programmer response
No action needed.

User response
No action needed.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CICS Socket Interface and API

Module
EZACIC02

Routing code
1

Descriptor code
2
Automation

This message is sent to the system console and to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example

The following is an example of the messages displayed when the stack has ended while the listener was processing data.

```
EZY1369E 01/10/06 12:59:32 LISTENER CSKL 10295 IS DELAYED, STACK TCPCS IS UNAVAILABLE
EZY1363I 01/10/06 12:59:33 LISTENER CSKL 10295 HAD 5 THREADS ACTIVE WHEN STACK TCPCS ENDED
EZY1367I 01/10/06 12:59:33 SOCK# IP ADDRESS PORT CHILD
EZY1368I 01/10/06 12:59:33 2 10.11.1.2 10245 PAYR
EZY1368I 01/10/06 12:59:33 12 2001:DB8:10::11:2:1 21089
EZY1368I 01/10/06 12:59:33 15 10.91.1.1 10245 INVN
EZY1368I 01/10/06 12:59:33 19 10.81.1.1 21212 ACCT
EZY1368I 01/10/06 12:59:33 999 2001:DB8:10::11:1:2 00901 ORDR
```

EZY1370I  mm/dd/yy hh:mm:ss LISTENER transactionid NUMSOCK numsock IS EQUAL TO OR GREATER THAN MAXFILEPROC maxfileproc

Explanation

A listener startup run-time check determined that the z/OS UNIX System Services MAXFILEPROC value was less than or equal to the listener’s NUMSOCK value. The listener’s accept processing pauses when the number of sockets that are supported by this listener exceeds the MAXFILEPROC value. No new connections are accepted until the number of sockets that are supported by this listener is less than the MAXFILEPROC value.

In the message text:

**mm/dd/yy**

The date (month/day/year) of the message.

**hh:mm:ss**

The time (hours:minutes:seconds) of the message.

**transactionid**

The listener's transaction ID.

**numsock**

The number of sockets supported by this listener.

**maxfileproc**

The maximum number of descriptors for files, sockets, directories, and any other file-system objects that can be concurrently active or allocated by a single process.

System action

Processing continues.

Operator response

Contact the system programmer.

System programmer response

Perform one of the following actions:

- Set the NUMSOCK value to be less than the MAXFILEPROC value using either the EZACICD macro or the EZAC configuration transaction, and then restart the listener. See the information about configuring a CICS TCP/IP environment in z/OS Communications Server: IP CICS Sockets Guide for more information about using the EZACICD macro and the EZAC configuration transaction.

- Set the MAXFILEPROC value to be greater than the NUMSOCK value using the SETOMVS system command. See the SETOMVS command information in z/OS MVS System Commands for information about dynamically changing the MAXFILEPROC option that z/OS UNIX System Services is currently using.
User response
No action needed.

Problem determination
Not applicable.

Source
z/OS Communications Server TCP/IP: CICS Socket Interface and API

Module
EZACIC02

Routing code
10

Descriptor code
12

Automation
This message is sent to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example
EZY1370I 01/19/06 10:07:33 LISTENER CSKL NUMSOCK 2000 IS EQUAL TO OR GREATER THAN MAXFILEPROC 250

EZY1371E mm/dd/yy hh:mm:ss AUTOMATIC APPLDATA REGISTRATION FAILED FOR TRANSACTION=transactionid TASKNO=taskno ERRNO=errno

Explanation
The automatic registration of application data failed for the reason described by the errno value.

In the message text:

mm/dd/yy
The date (month/day/year) of the message.

hh:mm:ss
The time (hours:minutes:seconds) of the message.

transactionid
The listener's transaction ID.

taskno
The task number assigned by CICS.

errno
erro is the UNIX System Services return code. These return codes are listed in the sockets and sockets extended return codes (ERROUs) in z/OS Communications Server: IP and SNA Codes.

System action
The application continues.
Operator response
Contact the system programmer.

System programmer response
See the information about automatically registering application data in z/OS Communications Server: IP Programmer's Guide and Reference for information about the socket commands affected by the automatic registration of application data.

_errno is the UNIX System Services return code. See the sockets and sockets extended return codes (ERRNOs) information in z/OS Communications Server: IP and SNA Codes for the action that you should take based on the SIOCSAPPLDATA IOCTL socket command return code.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: CICS Socket Interface and API

Module
EZACIC01, EZACIC02

Routing code
10

Descriptor code
12

Automation
This message is sent to the CICS transient data queue that is specified by the IP CICS Sockets ERRORTD configuration option.

Example
EZY1371E 07/01/06 10:07:33 AUTOMATIC APPLDATA REGISTRATION FAILED FOR TRANSACTION= CSKL TASKNO= 00000022L ERRNO= 55

EZY1372W Dataset *.TCPIP.DATA not found

Explanation
The system could not find a *.TCPIP.DATA data set to parse during initialization of the REXX socket interface.

System action
The program continues and attempts to use the default information.

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

**Module**
SOCMVS

**Procedure name**
INITFWRN

**Explanation**
The indicated device is allocating the specified data set for task-related storage.

**System action**
Rexx Sockets continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
SOCMVS

**Procedure name**
main

**Explanation**
The indicated task was unsuccessful. The **INFO=**\(\text{dynalloc\_info}\), **ERROR=**\(\text{dynalloc\_error}\) portion of the message appears only when **task** is ALLOCATE to indicate dynamic allocation. **task** is the name of the task that failed.

**System action**
Rexx Sockets continues.
Operator response
Notify the system programmer of the error.

System programmer response
Use the return code displayed in this message and the list of return codes in z/OS Communications Server: IP and SNA Codes to determine the cause of the error and respond as indicated. See the z/OS MVS Programming: Authorized Assembler Services Guide for information about information reason codes and error reason codes from DYNALLOC.

Module
SOCMVS

Procedure name
main

EZY1383I  Dataset not found

Explanation
A requested data set was not accessible to the host.

System action
Rexx Sockets continues.

Operator response
Notify the system programmer of the problem.

System programmer response
Make sure that the data set is in storage accessible to the host.

Module
SOCMVS

Procedure name
main

EZY1384E  Return code rc from IRXSUBCM function

Explanation
The specified function failed while trying to get storage.

System action
Rexx Sockets halts.

Operator response
Increase the region size for the application.

System programmer response
None.
Module
SOCMVS

Procedure name
main

**EZY1385E** Unable to acquire LOC=BETWEEN GETMAIN storage for QSAM DCB and exits

**Explanation**
The Rexx Sockets failed while trying to get storage.

**System action**
Rexx Sockets halts. Application continues.

**Operator response**
Increase the region size for the application.

**System programmer response**
None.

Module
SOCMVS

Procedure name
main

**EZY1386E** Return code rc from GETMAIN SVC

**Explanation**
The Rexx Sockets failed while trying to get storage.

**System action**
Rexx Sockets halts. Application continues.

**Operator response**
Increase the region size for the application.

**System programmer response**
None.

Module
SOCMVS

Procedure name
main

**EZY1388E** Return code xx from IDENTIFY SVC
**Explanation**
The Rexx Sockets could not identify the entry point for TCPERROR.

**System action**
Rexx Sockets halts. Application continues.

**Operator response**
Contact system programmer.

**System programmer response**
Verify that Rexx Sockets has been installed correctly.

**Module**
SOCMVS

**Procedure name**
main

---

**EZY1389E Unable to acquire Dynamic Save Area storage**

**Explanation**
The function GETMAIN, which is used to acquire storage, was unsuccessful. No storage is allocated.

**System action**
Rexx Sockets halts.

**Operator response**
Increase the region size for the application and try again.

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

**Module**
SOCMVS

**Procedure name**
main

---

**EZY1391E Unable to initialize Global DSA**

**Explanation**
The host was unable to initialize the global dynamic storage area (DSA). This can occur due to lack of storage accessibility to the host.

**System action**
RXSOCKET halts. TCPIP continues.
Operator response
Notify the system programmer of the error.

System programmer response
Increase the size of the storage region available to the application.

Module
SOCMVS

Procedure name
main

EZY1400I  Unable to establish ESTAE exit; processing continues

Explanation
Rexx Sockets was unable to establish an Error State Exit.

System action
Rexx sockets continues.

Operator response
Notify the system programmer of the error.

System programmer response
Verify that Rexx Sockets was installed correctly.

Module
SOCMVS

Procedure name
main

EZY1401I  REXX/ SOCKETS anchor located via IRXSUBCM address

Explanation
This debug message displays the address of persistent storage used as the anchor.

System action
Rexx sockets continues.

Operator response
None.

System programmer response
None.
Abend code detected in REXX/Sockets at location

REXX/Sockets loaded at rexx sockets location; Failing PSW: psw

register numbers: register_values

REXX/Sockets detected an abend. The abend code and the offset within the routine that caused the abend are displayed, followed by the location of the REXX/Sockets program, the Program Status Word (PSW) at the time of the abend, and the value of the general purpose registers.

REXX/Sockets ends and returns control to the application.

Determine the cause of the abend by referring to z/OS MVS System Codes manual for an explanation of the System abend codes. For more information, see the messages that usually follow these messages and are generated by the TSO/REXX abend handler.

None.

Module

SOCMVS

Procedure name

STAEGOOD

service SERVICE CALL Socket call: socket_call

Return_value retval hex

Return_code retcode hex

Reason_code rsncode hex

These messages show the return information from the z/OS UNIX System Services used to implement the REXX sockets API when SOCKDEBUG is specified in the TCPIP DATA data set. The BPXxxxx service call is service and the name of that service is socket_call.

For example: EZY1421I BPx1SOC SERVICE CALL Socket call: socket

These services, and the associated return information, are documented in z/OS UNIX System Services Programming: Assembler Callable Services Reference.
**Operator response**
None.

**System programmer response**
None.

**Module**
RXSOCKET

**Procedure name**
TRACER

**EZY1870E** CANNOT LOAD module ROUTINE.

**Explanation**
An attempt to load the indicated module into virtual storage was unsuccessful. Control is returned to the operating system with an error code indicating that a LOAD failure occurred.

**System action**
The task initialization function is terminated.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Check that the identified module resides in a library that is accessible to the MVS platform code. See z/OS Communications Server: IP Configuration Reference for information about required library residence for TCPIP components.

**Module**
MVPMAIN

**Procedure name**
Mainline code

**EZY1876I** server STACK FUNCTIONS STARTED WITH PARAMETER profile.

**Explanation**
Informational message displayed every time a TCPIP server or TCPIP is invoked. The profile parameters for each started server are displayed.

**System action**
The system continues invoking TCPIP and required services.

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

**Module**
MVPOCM

**Procedure name**
Mainline code

---

**EZY1877I**  module STACK FUNCTIONS SHUTDOWN IS COMPLETE, RC = rc.

**Explanation**
This message displays on the operator console after shutdown has completed for the load indicated by module. If the function indicated by module terminated normally, then RC = 0. If the function indicated by module terminated abnormally, then RC = 200.

**System action**
The function indicated by module terminates.

**Operator response**
None.

**System programmer response**
None.

---

**Module**
MVPMAIN

**Procedure name**
Mainline code

---

**EZY1889I**  MVP I/O Interrupt from device number CSW=channel status word.  

**Explanation**
This message displays if the debug option is used. The MVS platform processed an I/O interrupt on the indicated device. If the device number displayed is 01, the device is a fake device.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.
**Procedure name**
ProcIO

**EZY1890I**  
MVP CheckUserID for user id, rc = rc, rsn = reason.

**Explanation**  
This message displays if the debug option is used. The MVS platform called the SAF interface to check a user ID. This message displays the user ID that was entered and a return and reason code. If the return code is not zero, the user ID is not valid.

**System action**  
TCPIP continues.

**Operator response**  
Check and make sure that the syntax of the user ID is correct and that the ID is set up in the MVS system. See your system administrator for help with this message.

**System programmer response**  
Assist the operator as required.

**Module**  
MVPUTIL

**Procedure name**  
CHECKUID

**EZY1891I**  
MVP CCW to start=ccw

**Explanation**  
This message displays if the debug option is used. The MVS platform channel command word (CCW) to be issued is displayed. CCWs have either of two different formats. The two formats do not differ in the information contained in the CCW but only in the arrangement of the fields within the CCW. The formats are designated format 0 and format 1. Format 0 CCWs can be located anywhere in the first 16Mb of main storage. Format 1 CCWs can be located anywhere in main storage.

**System action**  
TCPIP continues.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
MVPEXCP

**Procedure name**  
SCC1
EZY1892I MVP HaltIO to device device issued.

Explanation
This message displays if the debug option is used. The MVS platform halt I/O routine (MVPEXCPH) was invoked for the indicated device.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPEXCP

Procedure name
MVPEXCPH

EZY1894I MVP OCM Invoked Operator (or) Start (or) Stop (or) Modify (or) Unknown

Explanation
This message is issued if the debug option is used. Operator indicates that the operator communication routine is invoked. Start, Stop, and Modify specifies which command was requested. Unknown indicates that MVP does not recognize the command received.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPOCM

Procedure name
MVPOCM1

EZY1895I MVP DEBUG table entry: keyword= ON (or) OFF

Explanation
This message displays when the debug option is used. The indicated command and its switch setting are displayed.
**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

MVPOCM

**Procedure name**

Dval

**EZY1896I**  
MVP action Control Register 0 = value.

**Explanation**

This message displays if the debug option is used. The read control register 0 and the set control register 0 routines are called to update control register 0. These are part of a list of tasks required to disable the reception of virtual machine communication facility (VMCF). For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

*action* is the action the function performed against the value in Control Register 0. The action will be either Read or Set.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

MVPDOSM

**Procedure name**

READC0, SETC0

**EZY1897I**  
MVP Received Ext Interrupt Code=ext_code.

**Explanation**

This message displays if the debug option is used. MVP received the indicated external interrupt code.

**System action**

TCPIP continues.
EZY1898I MVP Setting Ext Interrupt Type=type Value=value

Explanation
This message is displayed if the debug option is used. The PC external interrupt routine was set. The interrupt type and the clock comparator value is displayed. Valid type is external timer at (X'1004').

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPIOINT

Procedure name
ProcExt

EZY1899I MVP Ext Interrupt subclass ext_code is disabled, C0=current control register 0.

Explanation
This message is displayed if the debug option is used. An external interrupt code was disabled and the current control register zero (CR0) is displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
**Module**
MVPIOINT

**Procedure name**
ProcExt

**EZY1900I** MVP Clock Comparator Interrupt overridden by a new one.

**Explanation**
This message displays if the debug option is used. The external interrupt has been replaced by a new interrupt with a later clock comparator value.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPIOINT

**Procedure name**
ProcExt

**EZY1901I** MVP Close failed, number I/Os outstanding for address.

**Explanation**
This is displayed if the debug option is used and the devices indicated have failed to close. This could occur as a result of the function FindODB not being able to find the I/O devices to shut down.

**System action**
TCPIP continues.

**Operator response**
Make sure that you have specified the correct physical or channel addresses of the I/O devices that you want to close.

**System programmer response**
Assist the operator as necessary.
**EZY1902I** MVP Close Halt failed, *number IOs for address Halt Rc=code.*

**Explanation**
This message is displayed if the debug option is used. This indicates that when an IO failed to close, the **HALT** function was called to stop the indicated addresses but the device did not stop processing.

**System action**
The I/O devices will fail to halt processing.

**Operator response**
Stop the indicated devices using normal or abnormal shut down procedures and restart the devices if necessary.

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

**Module**
MVPCLOSE

**Procedure name**
Mainline code

---

**EZY1903I** MVP Task did not close ddname *name*.

**Explanation**
This message displays if the debug option is used. The ddname indicated was not closed before the TCPIP task ended.

**System action**
TCPIP continues with termination.

**Operator response**
Make sure that the ddname is valid and that it exists in the volume defined to the MVS system. If you require help with this message, contact the IBM Software Support Center.

**System programmer response**
Assist the operator as required.

**Module**
MVPUTIL

**Procedure name**
DEBCHECK

---

**EZY1904I** MVP Application return *routine rc* to MVPTASK routine.

**Explanation**
This message displays if the debug option is used. MVPTASK was called to process the initial platform attached routine. The routine return code is displayed indicating a return from the task.
System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPTASK

Procedure name
MVPTASK

EZY1905I  MVP Diagnose Number DiagCode issued:

Explanation
This message displays if the debug option is issued. It indicates the diagnose 7C subfunction code requested. This message should be followed by message EZY1906I, which specifies the registers allocated for this function.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPDODG

Procedure name
IssueDia

EZY1906I  MVP Registers: Regs1 Regs2 Regs3 Regs4

Explanation
This message displays if the debug option is issued. It follows message EZY1905I or message EZY1929I. This message indicates the registers assigned for the functions described in either of the two preceding messages mentioned.

System action
TCPIP continues.

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

**Module**
MVPDODG

**Procedure name**
IssueDia

**EZY1907E** MVP Diagnose code ignored, not supported yet.

**Explanation**
The module responsible for the initial handling of the Diagnose functions simulated by the MVS platform code received a request specifying a Diagnose function code that was not present in the simulation support. The incorrect Diagnose function code is identified in the message text. Control is returned to the calling module with a condition code of 0 and an error reason code of 0.

**System action**
The invocation of MVPDODG is terminated immediately after issuing the message.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
This error message is caused either by a programming error in the calling module or by corruption of the parameter list passed to the Diagnose simulation supervisor. Gather all available supporting documentation, and contact the IBM Software Support Center.

**Module**
MVPDODG

**Procedure name**
Mainline code

**EZY1910E** MVP Diag7C subfunction code not supported.

**Explanation**
The module responsible for providing the simulation of the Logical Device Support Facility (LDSF) received an input parameter representing a Diagnose X'7C' subfunction code that was not valid. The message text identifies the incorrect subfunction code. The following subfunction codes are simulated by the MVS platform:

- Initiate (code 1)
- Accept (code 2)
- Present (code 3)
- Terminate (code 4)
- Terminate_All (code 5)
- Status (code 6)
- Break (code 100)

EZY1xxxx messages 229
See the appropriate VM System Facilities for Programming publication for information about the LDSF (under the subject area Diagnose X'7C'). The simulation provided by the MVS platform is used in conjunction with Telnet support. The subfunction code 1000 is an MVS-only code used to provide support for a line-mode Break.

**System action**

Control is returned to the calling Telnet server module (TNLDSFP) with a condition code of 1 and an error reason code of 1. The invocation of MVPDG7C is terminated immediately. The Telnet server module subsequently fails the request it was processing at the time of the error.

**Operator response**

Tell the system programmer about the error.

**System programmer response**

This error message is caused either by a programming error in the associated Telnet server module or by corruption of the parameter list passed to the LDSF simulation routine. Gather all available supporting documentation, and contact the IBM Software Support Center.

**Module**

MVPDG7C

**Procedure name**

Mainline code

**EZY1912I** MVP Diag7C Initiate-Accept followed by Status not supported.

**Explanation**

The module responsible for providing the simulation of the Logical Device Support Facility (LDSF) received an Initiate request with a parameter list specifying that an Initiate-Accept followed by a Status should be performed. This capability is not supported by the simulation facilities present in the MVS platform code.

**System action**

The invocation of MVPDG7C is terminated immediately after issuing the message, and control is returned to the calling Telnet server module (TNLDSFP) with a condition code of 3 and an error reason code of 3. The Telnet server module subsequently fails the Initiate request that it was processing at the time of the error.

**Operator response**

Tell the system programmer about the error.

**System programmer response**

Because specification of this processing sequence depends on the value passed in the high-order byte of the second input parameter, a programming error in the calling module is the most probable cause. Corruption of the parameter list passed to the LDSF simulation routine is also a possible cause. Gather all available supporting documentation, and contact the IBM Software Support Center.

**Module**

MVPDG7C

**Procedure name**

FINIT
**EZY1913E** MVP Diag7C Initiate-Logical Device specification not supported.

**Explanation**
The module responsible for providing the simulation of the Logical Device Support Facility (LDSF) received an Initiate request with a parameter list specifying that session initiation should be performed for a specific logical device address. This capability is not supported by the simulation facilities present in the MVS platform code.

**System action**
The invocation of MVPDG7C is terminated immediately after issuing the message, and control is returned to the calling Telnet server module (TNLDSFP) with a condition code of 3 and an error reason code of 4. The Telnet server module subsequently fails the Initiate request that it was processing at the time of the error.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Because specification of this processing option depends on the value passed in the high-order byte of the second input parameter, a programming error in the calling module is the most probable cause. Corruption of the parameter list passed to the LDSF simulation routine is also a possible cause. Gather all available supporting documentation, and contact that IBM Software Support Center.

**Module**
MVPDG7C

**Procedure name**
FINIT

**EZY1916I** MVP VTAM Request request not supported.

**Explanation**
This message displays if the debug option is used. The MVS VTAM interface received a request that is not supported. The command is ignored.

**System action**
TCPIP continues.

**Operator response**
Refer this message to the system programmer.

**System programmer response**
Find what is issuing the request and correct it. Contact the IBM Software Support Center if assistance is required.

**Module**
MVPVTWK

**Procedure name**
MVPVTWK
**EZY1917I**  
**MVP VTAM Send_Msg ignored, not supported.**

**Explanation**  
This message displays if the debug option is used. The MVS VTAM interface received a request for Send_Msg that is not supported. The request is not honored.

**System action**  
TCPIP continues.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
MVPVTWK

**Procedure name**  
MVPVTWK

---

**EZY1921I**  
**MVP VTAM asked to initiate token, not implemented.**

**Explanation**  
This message displays if the debug option is used. The MVS VTAM interface received a request to initiate a session. This request is not supported by TCP/IP. The session ID or token requested is displayed.

**System action**  
TCPIP continues. If the token does not already exist, TCPIP sends a terminate request to VTAM for this token.

**Operator response**  
Refer this problem to the system programmer.

**System programmer response**  
Identify what is sending the initiate request from VTAM to TCP/IP and stop it. Contact the IBM Software Support Center for assistance if required.

**Module**  
MVPVTWK

**Procedure name**  
MVPVTWK

---

**EZY1926I**  
**MVP Received LDSF Ext Int: ext_code, Flag=LDSF_byte130 Reason=LDSF_reason.**

**Explanation**  
This message displays if the debug option is used. The MVS platform received an LDSF type external interrupt. The interrupt code, flag and reason code are displayed.
System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPIOINT

Procedure name
ProcExt

EZY1927I MVP Attach of module failed with return code rc.

Explanation
The module responsible for attaching subtasks running under the MVS platform failed to attach the indicated module as a subtask of the platform. The return code from the ATTACH invocation is identified in the message text.

System action
MVPATT posts the entry control block (ECB), for which MVPCALL is waiting, with a completion code indicating that the ATTACH failed. It then returns to a WAIT state, waiting on its list of ECBs. MVPCALL returns to its caller with a return code indicating that the requested subtask creation was unsuccessful. The calling module subsequently fails the underlying request.

Operator response
Tell the system programmer about the error.

System programmer response
See the applicable MVS Application Development Reference: Services for Assembler Language Programs publication for information about the given ATTACH return code. If the error is because of an insufficient region size, make the appropriate adjustments, and restart the task. For return codes indicating logic errors, a programming error in either the MVS platform code or the module to be attached is probable. Gather all available supporting documentation, and contact the IBM Software Support Center. The following are the possible return codes and their descriptions:

Return Code Description
0 Success
4 Was issued in a specify task abnormal exit (STAE)
6 No more storage
12 Invalid exit address or invalid parameter list address
MVPATT

Mainline code

EZY1928I MVP Attached module successfully.

Explanation
The indicated module was attached successfully by the MVS platform.

System action
MVPATT posts the ECB, for which MVPCALL is waiting, with a successful completion code indicating that the ATTACH to the module was successful.

Operator response
None.

System programmer response
None.

Module
MVPATT

Procedure name
Mainline code

EZY1929I MVP Diagnose Number DiagCode returned CondCode code

Explanation
This message displays if the debug option is issued. The requested diagnose code and a condition code are displayed. This message will be followed by message EZY1906I, which indicates the registers accessed for this function.

System action
TCPIP continues.

Operator response
Refer this to the system programmer.

System programmer response
Check the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information regarding the condition code received, and respond accordingly. Make sure the diagnose code number is valid.

Module
MVPDODG
**EZY1935I** MVP Pwd Check failed password?????, rc = rc, rsn = reason.

**Explanation**
This message displays if the debug option is used. The MVS platform called the SAF interface to check the password of a user ID. The password was not valid. Access is denied.

**System action**
TCPIP continues.

**Operator response**
Make sure that you used the correct syntax for the password for the indicated user ID. If you still have problems, it could be that the password is not set up accordingly. Refer this to your system administrator for correction.

**System programmer response**
Assist the operator as required.

**Module**
MVPUTIL

**Procedure name**
CHECKPWD

**EZY1936I** MVP access req. access to data set name on vol for user id failed, Rc = rc, Rsn = code.

**Explanation**
This message displays if the debug option is used. Access to the requested data set failed. See z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about return codes.

**System action**
TCPIP continues.

**Operator response**
Respond as indicated by the return code. Reissue your access request.

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

**Module**
MVPUTIL

**Procedure name**
DEBUGMSG

**EZY1937I** MVP system subsystem unavailable. system functions will fail.
**Explanation**

During a scan of the Subsystem Communication Vector Table (SSCVT) control blocks, the platform code could not locate the SSCVT associated with the indicated subsystem.

**System action**

The task is terminated, and control is returned to the system with a task completion code of 600. When the error occurs in module MVPXI (implying a failure during the initialization of the Virtual Machine Communication Facility (VMCF) subsystem itself), subsystem initialization is terminated, and control is returned to the system with a completion code of 4000.

**Operator response**

Tell the system programmer about the error.

**System programmer response**

Check that the identified subsystem is created and initialized during system initialization. See Step 3: Configure VMCF and TNF in z/OS Communications Server: IP Configuration Guide for more information about starting VMCF and TNF. Check for previous operator messages that might indicate why the particular subsystem failed to initialize. See the appropriate System Messages book for information about possible error messages from subsystem processing. A programming error is the most probable cause of the problem if subsystem processing error messages occur. If the problem cannot be remedied by modifications to subsystem definitions, gather all available documentation, and contact the IBM Software Support Center.

**Module**

MVPMAIN, MVPXVI

**Procedure name**

LOCCVT

**EZY1940I**  
MVP VMCF Function function invoked.

**Explanation**

This message displays if the debug option is used. It indicates the virtual machine communication facility (VMCF) function passed to TCPIP.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

MVPDOVM

**Procedure name**

SENDVMCF

**EZY1945I**  
MVP Maximum number of MODIFY commands queued to SMTP
Explanation
Twenty MODIFY smtpprocname,SMSG commands are currently in the simple mail transfer protocol (SMTP) queue. Wait for SMTP to process the current SMSG commands before issuing more MODIFY smtpprocname,SMSG commands.

System action
SMTP continues.

Operator response
Attempt to issue the next MODIFY smtpprocname,SMSG command at a later time. If SMTP still does not accept MODIFY smtpprocname,SMSG commands, contact the system programmer.

System programmer response
Determine whether SMTP is currently processing a large JES spool file. If it appears that SMTP has stopped, take a dump of the SMTP address space and contact the IBM Service Center with the SMTP job log. See the information about diagnosing SMTP problems in z/OS Communications Server: IP Diagnosis Guide for more information.

User response
Not applicable.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: SMTP

Module
MVPOCM

Routing code
Output is returned to the user.

Descriptor code
Output is returned to the user.

Example
None.

EZY1946I  MVP SMSG syntax is not valid.

Explanation
The syntax specified on the MODIFY smtpprocname,SMSG command is not valid. Issue a MODIFY smtpprocname,SMSG,HELP command to see the valid options.

System action
SMTP continues.
**Operator response**
Correct the error and issue the command again. For a list of valid SMSG commands, issue the MODIFY smtpprocname,SMSG,HELP command.

**System programmer response**
Not applicable.

**User response**
Not applicable.

**Problem determination**
None.

**Source**
z/OS Communications Server TCP/IP: SMTP

**Module**
MVPOCM

**Routing code**
Output is returned to the user.

**Descriptor code**
Output is returned to the user.

**Example**
None.

**EZY1947I**  MVP Invalid command command.

**Explanation**
The MVS platform received a command that is not valid for an application. See the MODIFY command information in z/OS Communications Server: IP System Administrator's Commands for a list of valid commands and their descriptions.

**System action**
The application continues.

**Operator response**
Reissue your request using a valid command.

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

**Module**
MVPOCM
EZY1948I   MVP Default Application is now *name*.

Explanation
The new default application name (for either transparent or line mode) is displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPOCM

Procedure name
AnyL

EZY1950I   MVP IUCV Query: Rc=rc IpRcode=code IpAudit=name.

Explanation
This message displays if the debug option is issued. An inter-user communication vehicle (IUCV) function was called. A system return code, IP return code, and IP audit address are displayed in this departing message from an IUCV function.

System action
TCPIP continues.

Operator response
None.

System programmer response
Respond as indicated by the IP return code for help in resolving any errors. A list of IP error codes and system return codes and their descriptions can be found in the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
MVPDOIUC

Procedure name
Depart

EZY1951I   MVP VMCF terminated: Cannot function without TNF.
**Explanation**
During the Virtual Machine Communication Facility (VMCF) address space initialization processing, a scan of the Subsystem Communication Vector Table (SSCVT) control blocks was performed, but the code could not locate the SSCVT associated with the Termination Notification Facility (TNF) subsystem. (The VMCF and TNF subsystems provide simulations of facilities provided by the VM operating system for the MVS platform code.)

**System action**
The VMCF subsystem initialization is terminated and control is returned to the system with a completion code of 5000.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Verify that the Termination Notification Facility (TNF) subsystem is active. Check for previous operator messages that might indicate why the TNF subsystem failed to initialize. See Step 3: Configure VMCF and TNF in z/OS Communications Server: IP Configuration Guide for more information about starting VMCF and TNF.

See the appropriate System Messages book for information about possible error messages from subsystem processing. A programming error is the most probable cause of the problem if subsystem processing error messages occur. If the problem cannot be remedied by modifications to subsystem definitions, gather all available documentation, and contact the IBM Software Support Center.

**Module**
MVPXVI

**Procedure name**
Mainline code

**EZY1952I**  
**MVP SSI: IEFSSREQ Process Sysout failure, rc=rc.**

**Explanation**
This message displays if the debug option is used. The subsystem request process failed as indicated by rc.

In the message text:

- **rc**
  
  the rc value, minus a decimal value of 100, is the JES SSI return code.

**System action**
TCPIP continues.

**Operator response**
Refer any problems to the system programmer.

**System programmer response**
This message is used for debugging purposes only. For an explanation of the JES SSI return code, see the return code information for SSI Function Code 1 in the z/OS MVS Using the Subsystem Interface. Refer any questions to the IBM Software Support Center.

**Module**
MVPSSI
**Procedure name**
MVPSSI1

**EZY1953I**  MVP SSI: SSOBRETN Process Sysout failure, rc=rc.

**Explanation**
This message displays if the debug option is used. The subsystem interface failed as indicated by rc.

In the message text:

- **rc**
  
  the rc value, minus a decimal value of 1000, is the JES SSOB return code.

**System action**
TCPIP continues.

**Operator response**
Refer this problem to the system programmer.

**System programmer response**
This message is used for debugging purposes only. For an explanation of the JES SSOB return code, see the SSOBRETN return codes for SSI Function Code 1 in the z/OS MVS Using the Subsystem Interface. Refer any questions to the IBM Software Support Center.

**Module**
MVPSSI

**Procedure name**
MVPSSI1

**EZY1954I**  MVP SSI: Process Sysout rc=rc, no data sets to select.

**Explanation**
This message displays if the debug option is used. The subsystem interface request could not complete because no data sets were found to select.

In the message text:

- **rc**
  
  is the JES SSOB return code SSSOEODS (no more data sets to select).

**System action**
TCPIP continues.

**Operator response**
Make sure that you define a job data set for the subsystem interface to process. Make sure that you follow correct data set naming conventions as described in z/OS Communications Server: IP Configuration Reference.

**System programmer response**
Assist the operator if required.
EZY1955I       MVP SSI: Process Sysout rc=rc, Job not found.

Explanation
This message displays if the debug option is used. The subsystem interface could not locate a specified job name.

In the message text:

rc is the JES SSOB return code SSONJOB (job not found).

System action
TCPIP continues.

Operator response
Make sure you define a correct job name to the subsystem interface routine for SMTP.

System programmer response
Assist the operator if required.

EZY1956I       MVP SSI: Selected Job: number # job id.

Explanation
This message displays when the debug option is used. The MVS platform subsystem interface called the routine that gets a job and allocates a data set for SMTP. The indicated job number and ID for the obtained job are displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
**Procedure name**
MVPSSI1

**EZY1957I**  MVP SSI: DYNALLOC Return code(s) rcode(s).

**Explanation**
This message displays if the debug option is used. The MVS subsystem interface routine was called to dynamically allocate a data set. The indicated return codes are displayed. Any return code other than 0 indicates an error in allocating the data set.

**System action**
TCPIP continues.

**Operator response**
Refer any errors to the system programmer.

**System programmer response**
Data set naming conventions are normally the main cause of errors in the dynamic allocation routine. See z/OS Communications Server: IP Configuration Reference for more information about dynamically allocating data sets.

**Module**
MVPSSI

**Procedure name**
MVPSSI1

**EZY1958I**  MVP SSI: Allocate Job return rc job ddname.

**Explanation**
This message displays if the debug option is used. The MVS platform subsystem interface was invoked to allocate a call. The job number, ddname, and return code are displayed. If the return code is nonzero, the call is not allocated.

**System action**
TCPIP continues.

**Operator response**
See the system programmer if the return code is nonzero.

**System programmer response**
If the return code is 8, this could indicate that the job number, ddname, or both are not valid. Check the job number or ddname and make sure that they are identified correctly to the subsystem interface. Correct the error and reinitiate the subsystem interface. If you continue to have problems with this message, contact the IBM Software Support Center.

**Module**
MVPSSI
**EZY1959I**  
MVP SSI: Free Job return rc job action.

**Explanation**
This message displays if the debug option is used. The MVS platform subsystem interface was invoked to free a call. The job number, action requested, and return code are displayed. If the return code is nonzero the call is not freed.

**System action**
TCPIP continues.

**Operator response**
See the system programmer if the return code is nonzero.

**System programmer response**
If the return code is 8, this could indicate that the action requested, job number, or both are not valid. Correct the error and reinitiate the subsystem interface. If you continue to have problems with this message, contact the IBM Software Support Center.

**Module**
MVPSSI

**Procedure name**
MVPSSI2

---

**EZY1960I**  
MVP SSI: MVPSSIW, state received.

**Explanation**
This message displays if the debug option is used. The MVS subsystem interface that waits for a JES file was called. The *state* of the call could be **Enable**, which specifies to look for a file or **Disable**, which specifies not to look for a file. If the state indicated is **Bad Post**, then an error occurred while posting the event control block (ECB).

**System action**
TCPIP continues.

**Operator response**
Refer any problems to the system programmer.

**System programmer response**
If the message indicates a bad post, you will need to gather all source information and any dumps that might be required and contact the IBM Software Support Center for help.

**Module**
MVPSSIW
**Procedure name**
MVPSSIW

**EZY1961I**   MVP SSI: MVPSSIW, Got Job job to handle.

**Explanation**
This message displays if the debug option is used. The MVS platform subsystem interface got the job indicated and will continue processing.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPSSIW

**Procedure name**
MVPSSIW

**EZY1962I**   MVP SSI: MVPSSIW, No Job to handle.

**Explanation**
This message displays if the debug option is used. The MVP platform subsystem interface was called to determine if there were any jobs to process, but no jobs were found.

**System action**
TCPIP continues.

**Operator response**
Normally, this message does not reflect an error situation, but only indicates that there are no jobs to process. If errors occurred during subsystem interface processing, messages EZY1952I–EZY1955I might have been issued to report the error. If there are jobs to process, make sure that you have identified a correct job number to the subsystem interface. Reinitiate the interface after correcting the problem.

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

**Module**
MVPSSIW

**Procedure name**
MVPSSIW
EZY1963I MVP SSI: Query called jobid Rcode=queryrc. (or) Allocate called jobid alocddn Rcode=allocrc. (or) Free called jobid action Rcode=freerc.

Explanation
This message displays if the debug option is used. The RDRQUERY, RDRALLOC, or the RDRFREE function is called to select, allocate, or free a job's data set for the subsystem interface. A job ID and a return code are returned for each function. The data set name is displayed when the allocate function is called, and an action indicating whether the data set has been freed is also displayed.

System action
TCPIP continues.

Operator response
Refer any errors to the system programmer.

System programmer response
See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for information about return codes, and respond accordingly.

Module
MVPDOSI

Procedure name
RDRQUERY, RDRALLOC, RDRFREE

EZY1964I MVP SSI: Interrupt called 0 (or) 1 Rcode=rc.

Explanation
This message is issued when the debug option is used. The RDRINT routine was called. This routine turns on or off the ability for the subsystem interface to accept interrupts from the MVS platform. Any return code other than 0 indicates a problem with this routine.

System action
TCPIP continues.

Operator response
Refer any errors to the system programmer.

System programmer response
See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for information about return codes. Respond to any bad return codes as indicated.

Module
MVPDOSI

Procedure name
RDRINT

EZY1965I MVP SSI: MVPSSIW, wait for file task started|ended.
Explanation
This message displays if the debug option is used. This message marks the start or the end of the MVS subsystem interface that waits for a JES file.

System action
Task ends. TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPSSIW

Procedure name
MVPSSIW

EZY1966I MVP Racf R15 = rc, SAFPRRET = rc, SAFPRREA = reason.

Explanation
This message displays if the debug option is used. The MVS system called the SAF interface to define a resource profile for a user ID and data set combination. A return code, system return code, and reason code are displayed.

System action
TCPIP continues.

Operator response
Respond as indicated by the return and reason codes. For more information about return codes, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference. For more information about RACF see z/OS Security Server RACROUTE Macro Reference. If you need additional help with this message, call the IBM Software Support Center.

System programmer response
Assist the operator as required.

Module
MVPUTIL

Procedure name
CHECKDSN

EZY1967I MVP RACSTAT Rc=rc.

Explanation
This message displays if the debug option is used. The MVS platform called the SAF interface to check if a security product is active. A return code of 0 indicates that the product is active. A return code other than 0 indicates that the product is not active.
System action
TCPIP continues.

Operator response
If the security product should be active and it is not, refer this to the system programmer.

System programmer response
Start the security product or perform an analysis on the product for information about why the product is not active.

Module
MVPUTIL

Procedure name
DEBUGMSG

EZY1968I     MVPAUTH: Getmain Rc=code

Explanation
This message is displayed if the debug option is used. The function Getmain which gets storage for the allocation of data sets has returned with the indicated return code, specifying that storage is not available. The data set function could not be processed.

System action
TCPIP continues.

Operator response
None.

System programmer response
See z/OS MVS Programming: Authorized Assembler Services Reference EDT-IXG for any help in resolving return codes. None.

Module
MVPAUTH

Procedure name
Internal_Auth_Check

EZY1969I     MVPAUTH: Freemain Rc=code

Explanation
This message is displayed if the debug option is used and the indicated data set in message EZY1970I could not be located. Storage might have been freed.

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

**System programmer response**
Respond as indicated by the return code. For information about MVS return codes affecting the Freemain function, see z/OS MVS Programming: Authorized Assembler Services Reference EDT-IXG.

**Module**
MVPAUTH

**Procedure name**
Internal_Auth_Check

**EZY1970I**    MVPAUTH: DSN=<data set name> Locate Rc=<code>

**Explanation**
This message is displayed if the debug option is used and the indicated data set could not be located. If the return code is zero, then the indicated data set has been located.

**System action**
TCPIP continues.

**Operator response**
Check the syntax of the data set requested and make sure that it is a valid data set. Reissue your request with the correct data set name.

**System programmer response**
None.

**Module**
MVPAUTH

**Procedure name**
Internal_Auth_Check

**EZY1972I**    MVP VMCF Function function: Rc=<return code>

**Explanation**
This message displays if the debug option is used. Displayed is the VMCF function called by TCPIP and the resulting return code.

**System action**
TCPIP continues.

**Operator response**
None.
**EZY1973I** MVP IUCV: *function* invoked.

**Explanation**
This message displays if the debug option is issued. An inter-user communication vehicle (IUCV) function was called.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

---

**EZY1974I** MVP Issued Command: Rc=rc R0=r0 “command text”.

**Explanation**
This message displays if the debug option is used. The MVS system received the indicated command. A return code and register 0 data (address space ID) are returned. If the return code is not zero, the command failed for START command.

**System action**
TCPIP continues.

**Operator response**
If the return code is not zero, check the syntax of the command issued and make sure it is correct. Make sure that the command is valid for the MVS system. If you continue to experience problems with this message, contact the IBM Software Support Center.

**System programmer response**
Assist the operator as required.
Module
MVPUTIL

Procedure name
Do_Command

**EZY1975I**  Attached task at address *address* starting.

**Explanation**
This message displays if the debug option is used. The asynchronous caller routine was called to attach a task. This message indicates that the asynchronous caller routine will attach a task for the routine with entry point of address to run under.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

Module
MVPCALI

Procedure name
MVPCALI

**EZY1976I**  *load module name* cannot Stop until started, try later.

**Explanation**
This message is issued if the debug option is used. The MVS platform received a command to stop the indicated load module, but it has not been started.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

Module
MVPOCM

Procedure name
MVPOCM1
**EZY1977I**  TaskExist rc task rc for user user id Asid taskasid

**Explanation**
This message displays if the debug option is used. The routine to check if a task exists is called to check if the requested task is available. The return code, user ID, and address space ID are displayed.

**System action**
TCPIP continues.

**Operator response**
Refer any errors to the system programmer.

**System programmer response**
See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for any information about error codes, and respond accordingly.

**Module**
MVPDOSY

**Procedure name**
TASKEXIS

**EZY1978I**  MVP Error setting Ext Interrupt STIMER(M) rc=rc

**Explanation**
This message displays if the debug option is used. The MVS platform received an error setting up an external interrupt or the interrupt type is not valid.

**System action**
TCPIP continues.

**Operator response**
Refer any problems to the system programmer.

**System programmer response**
The following table shows the possible return codes and their descriptions:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Request completed with problems.</td>
</tr>
<tr>
<td>8</td>
<td>Request could not be performed.</td>
</tr>
<tr>
<td>12</td>
<td>Error detected in an input parameter.</td>
</tr>
<tr>
<td>16</td>
<td>A system error occurred while performing the requested function.</td>
</tr>
<tr>
<td>99</td>
<td>Invalid interrupt type.</td>
</tr>
</tbody>
</table>

Respond to the return code as indicated. Call the IBM Software Support Center for additional help with this error.
**Module**
MVPEXTR

**Procedure name**
MVPEXTR

---

**EZY1979I**  SMTP cannot accept the MODIFY command - try again later

**Explanation**
The MVS platform received a command to modify SMTP, but the command was not accepted because SMTP initialization had not completed.

**System action**
SMTP continues.

**Operator response**
Issue the command after SMTP initialization completes.

**System programmer response**
No Action Needed.

**User response**
Not applicable.

**Problem determination**
Not applicable.

**Source**
z/OS Communications Server TCP/IP: SMTP

---

**Module**
MVPOCM

**Routing code**
10

**Descriptor code**
12

**Automation**
This message is sent to the console where the command was entered. Automation can be used to issue the MODIFY command at a later time.

**Example**
```
F SMTP,MSG,ST
EZY1979I SMTP cannot accept the MODIFY command - try again later
```

**EZY1980E**  module terminating due to previous error(s).
**Explanation**
The initialization of the application indicated by the module value failed as a result of unrecoverable errors. This message is issued for one of the following reasons:

- If this message is preceded by message number EZY1870, then the MVPTASK load module could not be loaded.
- If this message is preceded by message number EZY1981, then there was insufficient storage available.
- The VMCF and TNF subsystems are not active.

**System action**
The application ends.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Perform one of the following actions based on the reason for which this message was issued:

- If this message is preceded by message number EZY1870, verify that the MVPTASK load module is located in one of the load libraries accessible to the application.
- If this message is preceded by message number EZY1981, determine why there is insufficient storage for the application and increase storage, if necessary.
- Ensure that VMCF and TNF are active. To start VMCF and TNF as non-restartable subsystems, ensure that entries are defined for them in your IEFSSNxx PARMLIB member. To start VMCF and TNF as restartable subsystems, use the MVS started procedure EZAZSSI. See Step 3: Configure VMCF and TNF in z/OS Communications Server: IP Configuration Guide more information about starting VMCF and TNF.

**Module**
MVPMAIN

**Procedure name**
Mainline code

**EZY1981I** task insufficient storage.

**Explanation**
An attempt to obtain virtual storage in Subpool 73 for I/O completion and attention work areas was unsuccessful.

**System action**
The task is terminated and control is returned to the system with a task completion code of 8xx, with the xx portion reflecting the GETMAIN return code.

**Operator response**
Tell the system programmer about the error.

**System programmer response**
Determine if an increase in the region size of the started task will alleviate the problem. If the region size is not the problem, the most probable cause is a programming error involving a failure to release storage. Gather all available documentation and report the error to the IBM Software Support Center.
Module
MVPMAIN

Procedure name
Mainline code

**EZY1982E**  Terminating: one task is already active.

**Explaination**
To prevent the initiation of multiple TCPIP address spaces for the same task, the MVS platform code uses the ENQ macro to serialize the use of TCPIP resources. At task initialization, MVPMAIN issues an ENQ macro with a resource name of TCPIPSYS.task_name, requesting exclusive use. The message indicates that the ENQ failed.

**System action**
The task is terminated and control is returned to the system with a task completion code of 9xx, with the xx portion reflecting the ENQ return code.

**Operator response**
If the error was an unintentional attempt to start a TCPIP task that was already active, no actions are required. Otherwise, tell the system programmer about the error.

**System programmer response**
If the error was not because of an unintentional attempt to start a duplicate TCPIP task, examine the task completion code provided when the task was terminated. It will be of the form 9xx, with the xx portion of the code being the ENQ return code. See the appropriate Application Development Reference: Services for Assembler Language Programs publication for information about interpreting the return code. Follow the recommended problem resolution procedures indicated by the appropriate MVS Diagnosis book.

**Note:** The ENQ was issued with the RET=USE parameter.

If problem determination indicates that the MVS platform is in error, gather all available documentation and report the error to the IBM Software Support Center.

Module
MVPMAIN

Procedure name
Mainline code

**EZY1984I**  DIAG98: UnLock Page: Virtual=address

**Explanination**
This message displays if the debug option is used. The MVS system called the DIAG98 subroutine to unlock a page in central storage. The virtual address is displayed.

**System action**
TCPIP continues.

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

Module
MVPUTIL

Procedure name
MVPUTIL

EZY1985I  DIAG98: DoDiag98: Device=address CCWP=virtual channel control word CCWreal=real channel control word

Explanation
This message displays if the debug option is issued. The DoDiag98 subroutine of the Diag7C routine was called to perform I/O functions for a device with real channel control words (CCWs) that have been translated to virtual CCWs. The device address, virtual CCWs, and real CCWs are displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPDOIO

Procedure name
DoDiag98

EZY1986I  DIAG98: PgFix: Vaddr=address CC=code

Explanation
This message displays if the debug option is used. The DIAG98 subroutine was called to fix a page in the MVS system. The virtual address and completion code are returned.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
EZY1987I   DIAG98: PgFree: Vaddr=address CC=code

Explanation
This message displays if the debug option is used. The DIAG98 subroutine was called to free a previously fixed page of storage in the MVS system. The virtual address and completion code are provided.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

EZY1988I   DIAG98: UnLock All: FXBaddr=address FXBnext=address Vaddr=address

Explanation
This message displays if the debug option is used. The MVS platform system called the diagnose 98 subroutine to free a whole chain of virtual address. The current fixed page, next fixed page, and the virtual address are displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
**EZY1989I**  MVP Logical Unit name is now active to TCP/IP.

**Explanation**
The displayed logical unit is now active.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPOCM

**Procedure name**
AnyL

---

**EZY1990I**  MVP Logical Unit name is now inactive to TCP/IP.

**Explanation**
This message displays in response to your command to deactivate a logical unit to TCPIP. The LU is now inactive.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPOCM

**Procedure name**
AnyL

---

**EZY1991I**  MVP Logical Unit name is inactive to TCP/IP.

**Explanation**
This message displays in response to a status request. The LU name specified is not active to TCPIP.

**System action**
TCPIP continues.
Module
MVPOCM

Procedure name
AnyL

EZY1992I  MVP Logical Unit name is active to TCP/IP and in use.

Explanation
This message displays a response to a status request. The LU name indicated is currently active and in use.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPOCM

Procedure name
AnyL

EZY1993I  MVP Logical Unit name is active to TCP/IP and not in use.

Explanation
This message displays in response to a status request. The LU name specified is active to TCPIP but currently not in use.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
EZY1994E  MVP Logical Unit name not inactivated – in use to TCP/IP. User must issue LUINACT FORCE, luname to deactivate.

Explanation
This message indicates that the MVS platform received a request to deactivate an LU to TCPIP that failed. The LU remains active.

System action
TCPIP continues.

Operator response
Make sure that the LU is no longer in use and, as the message indicates, you must issue “LUINACT FORCE” followed by the LU name to deactivate it.

System programmer response
Assist the operator as required.

EZY1995E  MVPEXCP: MVPMAIN has Diag98 support but load module does not.

Explanation
This message displays if the debug option is used. The mainline code (MVPMAIN) has support for Diag98, or real channel control words (CCWs), but the PC load module indicated does not. Messages EZY1996E and EZY1997E will follow this message with more information.

System action
TCPIP continues.

Operator response
Refer this message to the system programmer.

System programmer response
Verify that the correct version of the load module is being used. Call the IBM Software Support Center for help if necessary.
**Procedure name**
MVPExCP

**EZY1996E**  Continuing using VCCWS (virtual CCWs) option.

**Explanation**
This message should be preceded by message EZY1995E, which indicates that a conflict exists between the load module and the mainline code (MVPMAIN) regarding support for the Diag98 function. The installation will continue, but it will use virtual channel control words (CCWs) instead of real CCWs.

**System action**
TCPIP continues.

**Operator response**
Refer this problem to the system programmer.

**System programmer response**
See message EZY1995E for help with the message.

**Module**
MVPDODG

**Procedure name**
Query98

**EZY1997E**  Installation was not done correctly.

**Explanation**
Your installation was not done correctly. See messages EZY1995E and EZY1996E, which precede this message for more information.

**System action**
TCPIP continues.

**Operator response**
Refer this problem to the system programmer.

**System programmer response**
See previous messages and their descriptions.

**Module**
MVPDODG

**Procedure name**
QAmode, Query98

**EZY1998E**  DIAG98: MVPMAIN does not support real CCWs but load module name does.
Explanation
The mainline code (MVPMAIN) does not have support for Diag98, or real channel control words (CCWs), although the load module indicated does. Messages EZY1996E and EZY1997E should follow this message.

System action
TCPIP continues.

Operator response
Refer this problem to the system programmer.

System programmer response
Check the mainline code (MVPMAIN) and make sure that it is running the code with support for Diag98 as defined in this source module (MVPDODG). Contact the IBM Software Support Center if you need more help with this message.

Module
MVPDODG

Procedure name
Query98

EZY1999E  QAmode: MVPMAIN and load module name are not at the same level.

Explanation
This message displays if the MVPMAIN code, which functions as a mainline entry for the MVS platform, does not have the same level of code for QAMODE support as the load module displayed. This message precedes message EZY1997E, which indicates that your installation was not done correctly.

System action
TCPIP continues.

Operator response
Refer this problem to the system programmer.

System programmer response
Check the mainline code (MVPMAIN) and make sure that it is running the code with support for QAMODE as defined in this source module (MVPDODG). Contact the IBM Software Support Center if you need more help with this error.

Module
MVPDODG

Procedure name
QAmode
Chapter 4. EZY2xxxx messages

EZY2000I  OCM General Default Application not configured for Transparent Mode.

Explanation
This indicates that the request to change the default application could not be honored because the new application is not configured for Transparent mode.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPOCM

Procedure name
AnyL

EZY2001I  OCM General Default Application not configured for Line mode.

Explanation
This indicates that the request to change the general default application could not be honored because the new application is not configured for line mode.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPOCM

Procedure name
AnyL

EZY2016I  MVP Diag7C LocAppl: Unknown function request function for ApplName application
Explanation
The locate application subroutine of the Diagnose 7C routine has returned with an unknown function request for the indicated application. The LocAppl subroutine checks the application request against the application table, declared in the MVPAPPL source code data set and defined in your hlq.PROFILE.TCPIP data set, to determine if the function request is allowed for the specified application.

System action
TCPIP continues. The specified request is not processed.

Operator response
Refer this problem to the system programmer.

System programmer response
The function flags that are passed for these applications are either restricted (1) or disconnectable (2). Make sure that the function specified for the requested application is defined in your hlq.PROFILE.TCPIP data set. Reissue your request specifying a valid function for the application requested.

Module
MVPDG7D

Procedure name
LocAppl

EZY2032W  LU name lu name specified in the command command does not match defined LU name or LU group name.

Explanation
The LU name or LU group entered in the command was not found in the LU status tables.

System action
The command is not performed. TCPIP continues.

Operator response
Correct the LU name and resubmit the command.

System programmer response
None.

Module
MVPOCM

Procedure name
ANYL

EZY2034W  LU specified in command is already inactivated.

Explanation
The LU name specified for the LUINACT command was already inactivated.
System action
The command is not performed. TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPOCM

Procedure name
ANYL

EZY2035W  Modify command command Incomplete. It is ignored.

Explanation
The required keyword was not specified.

System action
The command is not performed. TCPIP continues.

Operator response
See z/OS Communications Server: IP Configuration Reference for more information about the MODIFY command.

System programmer response
None.

Module
MVPOCM

Procedure name
ANYL

EZY2036I  Modify command DEBUG debug state is accepted.

Explanation
A change to the DEBUG flag has been changed to the specified value. The new debug option setting is now in effect.

System action
TCPIP continues.

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

**Module**
MVPOCM

**Procedure name**
ANYL

**EZY2037W** Invalid modify command. DEBUG debug state is ignored.

**Explanation**
An incorrect DEBUG option has been specified as part of the MODIFY command. The Telnet server ignored the DEBUG request. The debug option is not changed.

**System action**
TCPIP continues.

**Operator response**
Check the DEBUG option specified, correct, and resubmit the MODIFY command.

**System programmer response**
Assist the operator as required.

**Module**
MVPOCM

**Procedure name**
ANYL

**EZY2038I** Modify command LU specified matches LUGROUP lu. Accepted.

**Explanation**
The modify command was accepted because the specified LU was found in the LU group table.

**System action**
The command is performed. TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPOCM
Procedure name
ANYL

EZY2039W  LU specified lu is already activated. Ignored.

Explanation
The LU name specified with the LUACT parameter was already activated.

System action
The command is not performed. TCPIP continues.

Operator response
Check the specified LU and if incorrect, correct and reissue the MODIFY command.

System programmer response
Assist operator as required.

Module
MVPOCM

Procedure name
ANYL

EZY2040I  SMSG: VMCF is not active on the system.

Explanation
The message indicates that the program attempted to set up a communication session using the Virtual Machine Communication Facility (VMCF). The system was unable to locate or save the address space of the VMCF communication vector table. The program failed to initialize the client interface to VMCF.

System action
The communication does not occur because VMCF address space could not be established. The program requires the address space to continue. The program sets a return code of 16, indicating a system error to the calling program, and terminates execution. The VMCF address display command is not issued.

Operator response
Tell the system programmer about the error.

System programmer response
Make sure that the communication vector table and VMCF are active.

Module
MVPXDISP, MVPXSMG

Procedure name
Mainline code

EZY2041I  SMSG: Userid not specified.
**Explanation**
TCPIP received an SMSG command for the VMCF interface. You did not specify the user program, address space, or batch job to receive the SMSG.

**System action**
TCPIP continues.

**Operator response**
Reissue the SMSG command specifying a user program, address space, or batch job. For more information about SMSG see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference, or z/OS Communications Server: IP Configuration Reference.

**System programmer response**
Assist the operator as required.

**Module**
MVPXSMSG

**Procedure name**
MVPXSMSG

**EZY2042I**  SMSG: Userid specified incorrectly.

**Explanation**
The MVS platform received an SMSG command for the VMCF interface. You specified a user program, address space, or batch job that was too long.

**System action**
TCPIP continues. The SMSG command is not honored.

**Operator response**
Reissue the SMSG command specifying a user program, address space, or batch job of the correct length. For more information about SMSG see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference, z/OS Communications Server: IP User's Guide and Commands, or z/OS Communications Server: IP Configuration Reference.

**System programmer response**
Assist the operator as required.

**Module**
MVPXSMSG

**Procedure name**
MVPXSMSG

**EZY2043I**  SMSG: No message specified.
Explanation
TCPIP received an SMSG command with no message specified. The SMSG command is used to send a special message (SMSG) to a user program, address space, or batch job. The message text is required.

System action
TCPIP continues. The SMSG command is not honored.

Operator response
Reissue the SMSG command specifying the text of the SMSG to be sent. If this message occurred as a result of issuing SMSG TRACE on TSO from where the FTP server resides, then you should issue "SMSG FTTPSERVE TRACE". For more information about SMSG, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference. If you need more help with this message, contact the IBM Software Support Center.

System programmer response
Assist the operator as required.

Module
MVPXSMSG

Procedure name
MVPXSMSG

EZY2045I SMSG: user not logged on.

Explanation
TCPIP received an SMSG command for a user that is not logged on.

System action
TCPIP continues. The SMSG command is not honored.

Operator response
Check and make sure the target is active and reissue the SMSG command. For more information about the SMSG command, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

System programmer response
Assist the operator as required.

Module
MVPXSMSG

Procedure name
MVPXSMSG

EZY2048I SMSG: user quiesced; SMSG ignored.

Explanation
The SMSG command sent to the indicated user is ignored because the user or program is not active.
System action
TCPIP continues.

Operator response
Make sure that the user or program is active and reissue the SMSG command. Contact your system administrator for help if required.

System programmer response
Assist the operator as required.

Module
MVPXSMSG

Procedure name
MVPXSMSG

EZY2053I  MVPXDISP: User userID Asid addrspaceid.

Explanation
This message is the first in a set of messages displaying information about a specific user ID. The user ID userID indicates the client ID, and the Address Space Identifier (asid) for that client is displayed.

System action
The system continues with message EZY2054I, displaying user information for this ID.

Operator response
None.

System programmer response
None.

Module
MVPXDISP

Procedure name
Mainline code

EZY2054I  MVPXDISP: Data @ user data address Sm=system mask Cr0=control register 0 Flags=flags.

Explanation
This message displays in response to the MVPXDISP command after message EZY2053I. The user data information is displayed.

System action
TCPIP continues.

Operator response
None.
EZY2055I  MVPXDISP: Client of interface.

Explanation
This message displays in response to the MVPXDISP command and follows message EZY2054I. This message can display up to four times, once for each possible interface. The following list displays the four possible interfaces that can display:

- VMCF address space
- SMSG
- VMCF
- IUCV

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPXDISP

Procedure name
MVPXDISP

EZY2056I  MVPXDISP: IUCV mask=mask, Pending Ctl=control, Appl=application.

Explanation
This message displays in response to the MVPXDISP command and follows message EZY2055I. Information about the IUCV application is displayed. The information displayed includes enable, control pending interrupts, and application pending interrupts masks for the client.

System action
TCPIP continues.

Operator response
None.
System programmer response
None.

Module
MVPXDISP

Procedure name
MVPXDISP

EZY2057I MVPXDISP: VMCF: Buf=address, Len=length, Flgs=flags user=user Key=auth key.

Explanation
This message displays in response to the MVPXDISP command. The user data is displayed. Included are the user external interrupt buffer address and length, user flags, and user authority.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPXDISP

Procedure name
MVPXDISP

EZY2058I MVPXDISP: IUCV: Connections=number, Max=number.

Explanation
This message displays in response to the MVPXDISP command and follows message EZY2059I. The number of IUCV connections and the maximum number of IUCV connections allocated for the client are displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
**Procedure name**
MVPXDISP

**EZY2059I** MVPXDISP: VMCF: Pending count=number Flags=flags.

**Explanation**
This message displays in response to the MVPXDISP command and follows message EZY2057I. The number of VMCF pending connections and the VMCF flags are displayed.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPXDISP

**Procedure name**
MVPXDISP

**EZY2060I** MVPISAQ : name1 name2 Header at address

**Explanation**
This message displays when the debug option is used. The header name and address are displayed for an initial storage area (ISA) queue header. This message displays with messages EZY2061I — EZY2063I.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPOCM

**Procedure name**
PRISAQ

**EZY2061I** MVPISAQ : Subpool number 1st Getmain count count

EZY2xxx messages 273
Explanation
This message displays when the debug option is used. The subpool number and the total number of storage areas allocated by GETMAIN are displayed. This message displays with messages EZY2060I, EZY2062I, and EZY2063I.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPOCM

Procedure name
PRISAQ

EZY2062I MVPISAQ : 2nd Getmain count count

Explanation
This message displays when the debug option is used. The count of overflow ISA areas allocated is displayed. This message displays with messages EZY2060I, EZY2061I, and EZY2063I.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPOCM

Procedure name
PRISAQ

EZY2063I MVPISAQ : Frame size size Max asked max size(max size)

Explanation
This message displays when the debug option is used. The default frame size and the largest storage area allocated are displayed in decimal and hexadecimal form. This message displays with messages EZY2060I — EZY2062I.
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPOCM

**Procedure name**
PRISAQ

---

**EZY2064I**  **MVPXDISP: User name not found.**

**Explanation**
This message displays in response to the MVPXDISP command. The user indicated could not be found by this utility.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPXDISP

**Procedure name**
MVPXDISP

---

**EZY2065I**  **MVPXDISP: IUCV: Ctl flags=flags Appl flags=flags.**

**Explanation**
This message displays in response to the MVPXDISP command and follows message EZY2058I. The IUCV control flags and the IUCV application flags are displayed.

**System action**
TCPIP continues.

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

Module
MVPXDISP

Procedure name
MVPXDISP

EZY2066I MVPXDISP: Index=asid, Sm=system mask, Cr0=data, Flags=flags, User=user.

Explanation
This message displays in response to the MVPXDISP command. VTAM user information is displayed, including the address space ID, system mask, control register 0 data, flags, and user name.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
MVPXDISP

Procedure name
MVPXDISP

EZY2080I MVPMAIN: Storage could not be obtained for TCPIP address CVT, Name=pc load module name, Rcode=rc

Explanation
This message indicates that there is not enough common storage available for the TCPIP communication vector table (CVT) control block. This will not directly impact TCPIP functions. Some cleanup might not be performed when TCPIP is stopped.

System action
TCPIP continues.

Operator response
Refer this message to the system programmer.

System programmer response
Investigate why below the line, CSA storage is not available. Restart TCPIP when more common storage becomes available. For more help contact the IBM Software Support Center.
**EZY2081I**  
MVPMAIN: Anchor control block could not be found for TCP/IP address space CVT

**Explanation**  
The TCP/IP address space CVT could not be located because the control block containing its address could not be found.

**System action**  
TCP/IP continues.

**Operator response**  
Refer this message to the system programmer.

**System programmer response**  
Look for abend in GETMAIN during TNF subsystem initialization. Correct that problem then restart TCP/IP after you IPL the MVS system. For more help contact the IBM Software Support Center.

**EZY2090I**  
DUMPING VTAMinfo tables, Table number = table

**Explanation**  
Information notifying you that TCP/IP is about to display the indicated VTAM tables. The following are the application tables used by VTAM:

**Table Number**  
**Description**

0  
All application tables

1  
Telnet application table

2  
VTAM log mode information table

3  
LU group table

4  
IP group table

5  
IP to LU map table

6  
Default application table
Application table

LU status table (for all LUs)

System action
The program continues.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2091I __

Explanation
Message separator.

System action
The program continues.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2092I TELNET id=applid

Explanation
Information notifying you that TCPIP is about to display the Telnet application table used by VTAM.

System action
The program continues.
Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2093I Mode Table

Explanation
Information notifying you that TCPIP is about to display the mode table used by VTAM.

System action
The program continues.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2094I Mt Ptr=table, ElemCount=counter

Explanation
Information specifying the allocated storage, pointed to in the table value, for the VTAM Telnet mode table. The counter value specifies the number of entries in the mode table.

System action
The program continues and displays each entry in the table, as specified in message EZY1095I.

Operator response
None.

System programmer response
None.
Module
MVPBLVT

Procedure name
Mainline code

**EZY2095I**  Mt Elem=counter, ModeName=name, Flag=flag, Model=model, Class=class, DeviceType=type

Explanation
Information for specific VTAM Telnet mode table entries. All table entries including number, the mode name, table name, and any flags are displayed.

System action
The program continues and displays each entry in the table.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

**EZY2096I**  LU Group Table

Explanation
Information notifying you that TCPIP is about to display the logical unit group (LUGROUPS) table used by VTAM. This table contains an entry for each group defined with the LUGROUP statement. Each entry contains the group name, and a set of LU names that belong to the group.

System action
The program continues, and displays the VTAM LU names groups definition table.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code
**EZY2097I**  Lg Ptr=table, ElemCount=counter, Size=bytes

**Explanation**
Information specifying the allocated storage, pointed to in the table value, for the VTAM logical unit group (LUGROUPS) table. The counter value specifies the number of elements in the mode table. The bytes value specifies the size of the entries in the mode table.

**System action**
The program continues and displays each entry in the table, as specified in message EZY2098I.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT

**Procedure name**
Mainline code

---

**EZY2098I**  Lg Elem=element, GrpName=groupID, AlocCnt=count, UsedCnt=count

**Explanation**
Information about a specific element in the VTAM logical unit group (LUGROUPS) table. The element value specifies the element number of the LU group table. The groupID value specifies the element number of the LU group table. The count values specifies the dynamic storage allocated and the storage used up by the LU group table.

**System action**
The program continues and displays each element in the LU group table.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT

**Procedure name**
Mainline code

---

**EZY2099I**  Lg Mem=LUmember, MemberName=Luname
Explanation
Information about a specific VTAM logical unit group (LUGROUPS) table. The **LUmember** value specifies the member of the LU group table. The **LUname** specifies the name for the member of the LU group table.

System action
The system continues, and displays all present members of the LU group table.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2100I  IP Group Table

Explanation
Information notifying you that TCPIP is about to display the Internet Protocol (IP) group table used by VTAM. This table contains an entry for each IP address and Subnet addresses defined with the **IPGROUP** statement. Each entry contains either an IP address or a subnet address/mask pair and the group name the address belongs to.

System action
The program continues.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2101I  **Ig** Ptr=table, ElemCount=counter, Size=bytes, SubnetIndex=index

Explanation
Information specifying the allocated storage, pointed to in the **table** value, for a new VTAM internet protocol group (IPGROUP) table. The **counter** value specifies the number of elements in the IP group table. The **bytes** value specifies the size of the entries in the IP group table. The **index** value specifies the start of IP subnets in the IP group table.
**System action**  
The program continues and displays each entry in the IP group table, as specified in messages EZY2102I, EZY2103I, and EZY2104I.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
MVPBLVT

**Procedure name**  
Mainline code

**EZY2102I**  
Ig Elem=counter, GrpName=name,

**Explanation**  
Information indicating the IP group name where the VTAM table entries reside.

**System action**  
The system continues and displays message EZY2103I with additional information related to this message.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
MVPBLVT

**Procedure name**  
Mainline code

**EZY2103I**  
Ig MemberAddr=network.subnet.host.local, IsSubnet=flag

**Explanation**  
The IP address information for a specific member of the IP group table entries is displayed. The flag indicates if the subnet mask address is either ON, in use, or OFF, not used. If the subnet mask address is used it will be displayed in message EZY2104I following this message.

**System action**  
The system continues displaying message EZY2104I if the subnet address is in use. Otherwise, the system continues displaying IP group member addresses.
**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT

**Procedure name**
Mainline code

**EZY2104I**  Ig Ip SubnetMask=

**Explanation**
This message appears only if the subnet mask is in use. The IP address information for the subnet mask of a IP group table member is displayed. The subnet mask address is in dotted-decimal notation. It is used as a mask, or a filter for the member address with which it is associated. The Subnet allows specification of IP address ranges and all the hosts on a sub-network. The Subnet is specified with a Subnet address and Subnet mask. The Subnet address is ANDed with the Subnet mask to remove any insignificant bits and stored as an IP address. The Subnet mask must be ANDed with the Telnet IP address and compared with the above Subnet address to determine if the IP address belongs to the subnet. The mask is stored after the Subnet address as an IP address.

**System action**
The system continues displaying IP group member addresses.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT

**Procedure name**
Mainline code

**EZY2105I**  IP to LU Map Table

**Explanation**
Information notifying you that TCPIP is about to display the map internet protocol (IP) address groups to VTAM logical unit (LU) names table. This table contains an entry for each mapping defined with the LUMAP statement. Each entry contains either an IP address or an IP group name that is mapped to a VTAM LU name or an LU group name.

**System action**
The program continues and displays the information.
Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

**EZY2106I**  Mp Ptr=table, ElemCount=counter, Size=bytes, AddrIndex=index

Explanation
Information specifying the allocated storage, pointed to in the table value, for a new VTAM internet protocol group (IP) to logical unit (LU) map table. The counter value specifies the number of elements in the IP to LU map table. The bytes value specifies the size of the IP to LU map table. The index value specifies the index to address a new part in the IP to LU map table.

System action
The program continues and displays each entry in the IP to LU map table, as specified in messages EZY2107I, EZY2108I, and EZY2109I.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

**EZY2107I**  Mp Elem=counter, LU id=id, IsGroup=groupflag, IsHost=hostflag

Explanation
Information for specific LU ID of the IP to LU map table. The table element number, and the LU identification number, id, are displayed. The elements can be a VTAM LU name or an LU group name. The groupflag is a flag used to identify if this LU element is a VTAM LU name or an LU group The hostflag is a flag used to identify if this LU element has a host IP address or a group name.

System action
The system continues displaying message EZY2108I if the this element is a host, or the system displays message EZY2109I to display the IP group name.
**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT

**Procedure name**
Mainline code

**EZY2108I**  Mp Ip Address=network.subnet.host.local

**Explanation**
This information is displayed only if the LU element is a host. The IP address information for a specific host of the IP to LU map table entries is displayed.

**System action**
The system continues displaying the IP to LU map table elements.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT

**Procedure name**
Mainline code

**EZY2109I**  Mp Ip GroupName =name

**Explanation**
This information is displayed only if the LU element is an IP group name. The group name of the IP to LU map element is displayed.

**System action**
The system continues displaying the IP to LU map table elements.

**Operator response**
None.

**System programmer response**
None.
**Explanation**

Information notifying you that TCPIP is about to display the default application (appl) table used by VTAM. This table contains an entry for each mapping defined with the DEFAULTAPPL or the LINEMODEAPPL statement. Each entry contains either an IP address, an IP group name, or a linkname that is mapped to an application name. This application becomes the default application when the Telnet user has not specified one.

The order of the search is IP address, IP group name, and finally the Linkname. There is also a special match-all entry that is used as the default if Telnet session has no default application explicitly defined.

**System action**

The system continues displaying the default application table.

**Operator response**

None.

**System programmer response**

None.

**Module**

MVPBLVT

**Procedure name**

Mainline code

**EZY2110I  Default Application Table**

**Explanation**

Information specifying the allocated storage, pointed to in the table value, for a new VTAM default application (appl) table. The counter value specifies the number of elements in the default appl table. The bytes value specifies the size of the default appl table. The index value specifies the default appl table index to the IP addresses part.

**System action**

The program continues and displays each entry in the default appl table, as specified in messages EZY2112I, EZY2113I, and EZY2114I.

**Operator response**

None.

**System programmer response**

None.
Module
MVPBLVT

Procedure name
Mainline code
EZY2112I Da Elem=counter, ApplName=name, LogMode=mode, IsHost=hostflag

Explanation
Information for specific entries of the default application (appl) table. The table element number, the VTAM appl name, as specified in VTAMLST, and the logon mode, are displayed. The logon mode mode can be either line or transparent mode. The hostflag variable is a flag used to identify if this default element has a host IP address.

System action
The system continues displaying message EZY2113I if this default entry has the IsHost variable set to ON. The true condition identifies this element as being a host, any other condition and the system displays the destination name in message EZY2109I.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code
EZY2113I Da Ip Address=network.subnet.host.local

Explanation
This information is displayed only if the default element is a host. The IP address information for a specific host of the default table entries is displayed.

System action
The system continues displaying the default table elements.

Operator response
None.

System programmer response
None.

Module
MVPBLVT
**Procedure name**
Mainline code

**EZY2114I**  Da Ip GroupName =*name*

**Explanation**
This information is displayed only if the default element is either an IP group name or a linkname. The group name of the default element is displayed.

**System action**
The system continues displaying the default table elements.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT

**Procedure name**
Mainline code

**EZY2115I**  Application Table

**Explanation**
Information notifying you that TCPIP is about to display the application (appl) table used by VTAM.
This table contains the applications which are permitted access through Telnet as specified by the RESTRICTAPPL or the ALLOWAPPL statement. The application entries contain an optional RestrictUsers table identifying the RESTRICTed applications, or an AllowLuNames set identifying the ALLOWed applications.

**System action**
The system continues displaying the application table.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT

**Procedure name**
Mainline code

**EZY2116I**  Ap Ptr=*table*, ElemCount=*counter*, Size=*bytes*
Explanation
Information specifying the allocated storage, pointed to in the table value, for a new VTAM application (appl) table. The counter value specifies the number of elements in the application table. The bytes value specifies the size of the appl table.

System action
The program continues and displays each entry in the appl table, as specified in the following messages:
- EZY2117I
- EZY2118I
- EZY2119I
- EZY2120I
- EZY2121I
- EZY2122I

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2117I  Ap Elem=counter, ApplName=name, ApplMask=X'mask'

Explanation
Information for specific entry of the application (appl) table. The table element number, the VTAM appl name, as specified in VTAMLST, and the appl comparison mask mask, are displayed. The appl mask can be used for logical comparisons to validate characters in the appl name and user IDs specified on the RESTRICTAPPL and the ALLOWAPPL statements.

System action
The system continues displaying message EZY2118I.

Operator response
None.

System programmer response
None.

Module
MVPBLVT
Procedure name
Mainline code

**EZY2118I**  Ap AlocCnt=count, UsedCnt=count, Discon=Discflag, Restrict=Rstrflag

**Explanation**
Information about a specific element in the VTAM application (appl) table. The count values specifies the dynamic storage allocated and the storage used up by the appl table. The Discflag allows an application to be disconnected by a user leaving the VTAM session active. The Rstrflag identifies this application as a restricted appl if it is set to the ON flag.

**System action**
The program continues and displays each element in the appl table. If the Rstrflag is set ON, then the program continues and displays the restricted applications in messages EZY2119I, EZY2120I, and EZY2121I.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT

---

**EZY2119I**  Ap Elem=counter, UserName=id, UserMask=X'mask'

**Explanation**
This information is displayed only if the Rstrflag is set to ON, in message EZY2118I, specifying a restricted table member. This information is for specific restricted users of the application (appl) table. The table element number, the user name ID, as specified in RESTRICTAPPL statement, and the user comparison mask mask, are displayed. The user ID mask can be used for logical comparisons to validate characters in the user IDs specified on the RESTRICTAPPL statements.

**System action**
The system continues displaying message EZY2120I and EZY2121I.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPBLVT
Procedure name
Mainline code

**EZY2120I**  Ap AlocCnt=\texttt{count}, UsedCnt=\texttt{count},

Explanation
This information is displayed only if the \texttt{Rstrflag} is set to \texttt{ON}, in message EZY2118I, specifying a restricted table member. This information is about a specific element in the VTAM application (appl) table. The \texttt{count} values specifies the dynamic storage allocated and the storage used up by the appl table.

System action
The program continues and displays each element in the appl table, and continues to message EZY2121I.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

**EZY2121I**  Ap Elem=\texttt{counter}, RestrictLuName=\texttt{name}

Explanation
This information is displayed only if the \texttt{Rstrflag} is set to \texttt{ON}, in message EZY2118I, specifying a restricted table member. This information is about a specific element in the VTAM application (appl) table. The \texttt{counter} values specifies the appl table entry. The \texttt{name} specifies the restricted user LU name for that entry in the appl table.

System action
The program continues and displays each restricted user LU names in the appl table.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

**EZY2122I**  Ap Elem=\texttt{counter}, AllowLuName=\texttt{name}
Explanation
This information is about a specific element in the VTAM application (appl) table. The counter value specifies the appl table entry. The name specifies the allowed user logical unit (LU) name for that entry in the appl table. This table contains the applications which are permitted access through Telnet as specified by the ALLOWAPPL statement.

System action
The program continues and displays each allowed user LU names in the appl table.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2130I LU Status Table

Explanation
Information notifying you that TCPIP is about to display the logical unit (LU) status table used by VTAM. The table is used to keep track of the status of VTAM LU names as they are used in Telnet connections.

System action
The system continues and displays messages EZY2131I, and EZY2132I.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2131I Ls Ptr=number, AlocCnt=count, UsedCnt=count

Explanation
Information specifying the allocated and used logical units pointed to in the logical unit (LU) status table used by VTAM.
System action
The system continues and displays message EZY2132I.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2132I  Ls Elem=number, Name=luname, Flags=Ls_Default flag

Explanation
Displays specific information about the entries in the logical unit (LU) status table. The LU element numbers number, names luname, and any optional flags flags are displayed.

System action
The system continues.

Operator response
None.

System programmer response
None.

Module
MVPBLVT

Procedure name
Mainline code

EZY2133E  VTAM UssTable table name can not be loaded.

Explanation
This message displays if the debug option is used. The VTAM unformatted systems service table indicated could not be loaded.

System action
TCPIP continues.

Operator response
Refer this message to the system programmer.
System programmer response
Make sure that the table name exists and that the program has access to it. Call the IBM Software Support Center for more help if necessary.

Module
MVPDG7D

Procedure name
MVPDG7DD

EZY2134I   STOP COMMAND IGNORED tcpipprocname ALREADY BEING STOPPED

Explanation
This message indicates that a subsequent stop command (P TCPIP) was received while TCPIP was processing a previous stop command. The subsequent stop command is ignored and TCPIP ends.

System action
TCPIP ends.

Operator response
None.

System programmer response
None.

Module
MVPOCM

Procedure name
MVPOCMII

EZY2142I   Hung VMCF Interrupt scheduled for ASID

Explanation
This message indicates that TCPIP has encountered a VMCF interrupt pending and the interrupt processing for the client is not running.

System action
TCPIP continues. The VMCF interface is hung.

Operator response
Refer this message to the system programmer.

System programmer response
Reinitiate the VMCF interface and reissue the interrupt if required. If the problem continues, contact the IBM Software Support Center.
Module
MVPXVI

Procedure name
MVPXVI

EZY2143I  Hung IUCV_A Interrupt scheduled for ASID

Explanation
This message indicates that TCPIP has encountered a IUCV_A interrupt pending and the interrupt processing for
the client is not running.

System action
TCPIP continues. The IUCV application is hung.

Operator response
Refer this message to the system programmer.

System programmer response
Restart the IUCV application and reissue the interrupt if required. If the problem continues, contact the IBM
Software Support Center.

Module
MVPXVI

Procedure name
MVPXVI

EZY2144I  Hung IUCV_C interrupt scheduled for ASID

Explanation
This message indicates that TCPIP has encountered a IUCV_C interrupt pending and the interrupt processing for
the client is not running.

System action
TCPIP continues. The IUCV interface is hung.

Operator response
Refer this message to the system programmer.

System programmer response
Reinitiate the IUCV interface and reissue the interrupt if required. If the problem continues, contact the IBM
Software Support Center.
**EZY2145I  VMCF cleanup for USERID id in ASID**

**Explanation**
This message indicates that VMCF is running cleanup for the indicated user ID.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

---

**EZY2370I  ssock_max number**

**Explanation**
This message displays the maximum number of simultaneous sockets available for use. This message is displayed when the debug trace option is specified.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

---

**EZY2371I  Initialized echo server**

**Explanation**
The echo server is initialized and ready to process incoming UDP requests.
System action
The application continues normal processing.

Operator response
To remove informational messages, start the echo server without any trace options. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for information about the echo server.

System programmer response
None.

Module
MISCSRV

Procedure name
init_socket

**EZY2372I**  Initialized TCP echo server

**Explanation**
The echo server is initialized and ready to process incoming TCP requests.

**System action**
The application continues normal processing.

**Operator response**
To remove informational messages, start the echo server without any trace options. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about the echo server.

**System programmer response**
None.

Module
MISCSRV

Procedure name
init_TCPsocket

**EZY2373I**  Initialized discard server

**Explanation**
The discard server is initialized and ready to process incoming UDP requests.

**System action**
The application continues normal processing.
**Operator response**
To remove informational messages, start the discard server without any trace options. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about the discard server.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
init_socket

<table>
<thead>
<tr>
<th>EZY2374I</th>
<th>Initialized TCP discard server</th>
</tr>
</thead>
</table>

**Explanation**
The discard server is initialized and ready to process incoming TCP requests.

**System action**
The application continues normal processing.

**Operator response**
To remove informational messages, start the discard server without any trace options. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about the discard server.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
init_TCPsocket

<table>
<thead>
<tr>
<th>EZY2375I</th>
<th>Initialized character generator server</th>
</tr>
</thead>
</table>

**Explanation**
The character generator server is initialized and ready to process incoming UDP requests.

**System action**
The application continues normal processing.

**Operator response**
To remove informational messages, start the character generator server without any trace options. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about the character generator server.
System programmer response
None.

Module
MISCSRV

Procedure name
init_socket

EZY2376I Initialized TCP character generator server

Explanation
The character generator server is initialized and ready to process incoming TCP requests.

System action
The application continues normal processing.

Operator response
To remove informational messages, start the character generator server without any trace options. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about the character generator server.

System programmer response
None.

Module
MISCSRV

Procedure name
init_TCPsocket

EZY2380I before selectex: readmask

Explanation
This message displays the contents of the readmask used by the function selectex to determine if a socket is prepared to read data. This message is displayed when the debug trace option is specified.

System action
The application continues.

Operator response
None.

System programmer response
None.

Module
MISCSRV
**Procedure name**
main

**EZY2381I** exiting

**Explanation**
This message is displayed to indicate that the miscellaneous server is closing all sockets and exiting the application.

**System action**
The miscellaneous server halts.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
main

**EZY2385I** after selectex: readmask

**Explanation**
This message displays the value of the readmask used by the function selectex to determine if a socket is ready to read data.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
main

**EZY2386I** socket number port number is condition
Explanation
This message indicates that data has been received over the indicated socket. This message is displayed when the debug trace option is specified.

System action
The application continues.

Operator response
None.

System programmer response
None.

Module
MISCSRV

Procedure name
main

EZY2387I before cextpost
EZY2388I ended interrupt handler: rc value

Explanation
The interrupt handler is being closed. The \textit{rc} portion of this message indicates whether the closing is successful. This message is displayed when the debug trace option is specified.

System action
The interrupt handler halts. The application continues.

Operator response
None.

System programmer response
None.

Module
MISCSRV

Procedure name
cextpost

EZY2389I UDP echo protocol

Explanation
A UDP echo request is being processed. The server echoes the data received. This message is displayed when the debug trace option is specified.
System action
The application continues.

Operator response
None.

System programmer response
None.

Module
MISCSRV

Procedure name
doecho

EZY2390I  Echo server: socket number address name

Explanation
This message displays the socket value and the address family (in dotted decimal notation) for the echo server. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about AF_INET or AF_IUCV.

System action
The application continues.

Operator response
None.

System programmer response
None.

Module
MISCSRV

Procedure name
doecho

EZY2391I  UDP discard protocol

Explanation
The UDP discard protocol is initialized. The UDP discard protocol will discard the data received by the MISCSRV.
System programmer response
None.

Module
MISCSRV

Procedure name
dodisc

EZY2392I Discard server: socket number address name

Explanation
This message displays the socket value and address (in dotted decimal notation) for the discard server.

System action
The application continues.

Operator response
None.

System programmer response
None.

Module
MISCSRV

Procedure name
dodisc

EZY2393I UDP character generator

Explanation
The UDP character generator is initialized. This will cause random data to return for each datagram that is received. This message is displayed when the debug trace option is specified. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about the character generator.

System action
The application continues.

Operator response
None.

System programmer response
None.

Module
MISCSRV
**Procedure name**
dochargen

**EZY2394I**  Character server: socket number address name

**Explanation**
This message displays the socket number and address (in dotted decimal notation) for the character server.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISSCRV

**Procedure name**
dochargen

**EZY2395I**  link name name family name home address name

**Explanation**
This message states the link name, family and home address of all the system adapter interfaces.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISSCRV

**Procedure name**
getifconf

**EZY2396I**  name socket number port number

**Explanation**
The link name, socket number, and well-known port number are displayed.
**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
init_socket

---

**EZY2397I**  
socket number port number

**Explanation**
The socket number and well-known server port number are displayed.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
init_TCPsocket

---

**EZY2398I**  
socket number af number

**Explanation**
The socket number and address family number, which indicates an AF_IUCV connection, are displayed for the SMSG socket.

**System action**
The application continues.

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

**Module**
MISCSRV

**Procedure name**
SMTPsocket

**EZY2399I**  TCP echo protocol

**Explanation**
A TCP echo request is being processed. The server echoes the data received. This message is displayed when the debug trace option is specified.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
doTCPecho

**EZY2401I**  TCP Discard protocol

**Explanation**
A TCP discard request is being processed. The server discards the data received. This message is displayed when the debug trace option is specified.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.
**EZY2403I** TCP Character Generator protocol

**Explanation**
A TCP character generator request is being processed. The character generator server creates random data in response to data received. This message is displayed when the debug trace option is specified.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**EZY2405I** Display SMSG

**Explanation**
An SMSG command is being processed. Shutdown and trace commands are the only supported SMSG commands for the miscellaneous servers. See the z/OS Communications Server: IP Configuration Guide and the z/OS Communications Server: IP User's Guide and Commands for more information about the SMSG command. This message is displayed when the debug trace option is specified.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**EZY2406I** Received message:
**Explanation**
This message header is followed by information pertaining to the status of the SMSG message requested.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
dispSMSG

---

**EZY2414I**  
shutting down

**Explanation**
The MISCSRV server is closing due to user or operator request. The server closes all connections and halts. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about the SHUTDOWN function.

**System action**
The application halts.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
dispSMSG

---

**EZY2416I**  
Unknown command: name

**Explanation**
The MISCSRV server received a command that it does not recognize. The command is ignored.

**System action**
The application continues.
**Operator response**
Reissue the command using the SHUTDOWN command or TRACE command. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about the SHUTDOWN or TRACE commands.

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

**Module**
MISCSRV

**Procedure name**
dispSMSG

**EZY2418I**  closed STREAM Socket: *number* rc: *number*

**Explanation**
The miscellaneous server closes the socket specified in the message.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
closesock

**EZY2419E**  error for select (*tcperror*)

**Explanation**
The select function was unsuccessful. (*tcperror*) is the number of the message where you can find more detailed information about this error.

**System action**
The application attempts to continue.

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

Respond as indicated *(tcperror)*. See the *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference* for more information about select().

**Module**

MISCSRV

**Procedure name**

selectex

---

**EZY2420E**  
**selectex (tcperror)**

**Explanation**

The selectex function was unsuccessful. *(tcperror)* is the number of the message where you can find more detailed information about this error.

**System action**

The application attempts to continue.

**Operator response**

Notify the system programmer of the error.

**System programmer response**

Respond as indicated in *(tcperror)*. See *selectex()* in the *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference* for more information.

---

**Module**

MISCSRV

**Procedure name**

selectex

---

**EZY2421E**  
**recvfrom() (tcperror)**

**Explanation**

The recvfrom function found no data while in nonblocking mode. *(tcperror)* is the number of the message where you can find more detailed information about this error.

**System action**

The application continues.

**Operator response**

Notify the system programmer of the error.

**System programmer response**

Respond as indicated in *(tcperror)*. For more information about the *(tcperror)* portion of this message, see the *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference*.  

---

EZY2xxx messages 311
Module
MISCSRV

Procedure name
doecho

EZY2422E sendto() (tcperror)

Explanation
The buffer space that is used to hold the data to be transmitted is inadequate, or no data was found in the buffer space. If the socket is in nonblocking mode a return code EWOULDBLOCK is set.
(tcperror) is the number of the message where you can find more detailed information about this error.

System action
The application continues.

Operator response
Notify the system programmer of the error.

System programmer response
Respond as indicated in (tcperror). See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information on the sendto() function.

Module
MISCSRV

Procedure name
doecho

EZY2424E create_TCP::socket (tcperror)

Explanation
An error occurred when trying to create a socket connection.
(tcperror) is the number of the message where you can find more detailed information about this error.

System action
The MISCSRV halts. The application continues.

Operator response
Notify the system programmer of the error.

System programmer response
Respond as indicated in (tcperror). For more information on the error condition see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
getifconf
**Procedure name**
init_TCPsocket

**EZY2425E**  
create_UDPC_port::socket *(tcperror)*

**Explanation**
The function create_UDPC_port, which is used to open a UDP connection, was unsuccessful. *(tcperror)* is the number of the message where you can find more detailed information about this error.

**System action**
The UDP application MISCSRV halts.

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

**System programmer response**
Respond as indicated in *(tcperror)*. For more information on the error condition see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
MISCSRV

**Procedure name**
init_socket

**EZY2426E**  
create_TCP_port::bind *(tcperror)*

**Explanation**
The bind() call encountered an error when trying the bind an address to the socket. *(tcperror)* is the number of the message where you can find more detailed information about this error.

**System action**
The application continues.

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

**System programmer response**
Respond as indicated in *(tcperror)*. See z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for a description of error values.

**Module**
MISCSRV

**Procedure name**
init_TCPsocket

**EZY2427E**  
create_UDPC_port::bind *(tcperror)*
**Explanation**
The bind() function encountered an error when trying the bind a UDP port.
(tcperror) is the number of the message where you can find more detailed information about this error.

**System action**
The application continues.

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

**System programmer response**
Respond as indicated in (tcperror). For more information about the bind() function and addition error values see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
MISCSRV

**Procedure name**
init_socket

---

**EZY2428E** create_TCP_port::listen (tcperror)

**Explanation**
The listen function, which is used to create a TCP connection to listen for data, encountered an error.
(tcperror) is the number of the message where you can find more detailed information about this error.

**System action**
The application continues.

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

**System programmer response**
Respond as indicated in (tcperror). For more information about the listen function and additional error values see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
MISCSRV

**Procedure name**
init_TCPsocket

---

**EZY2429E** create_IUCV_port::socket (tcperror)

**Explanation**
An error occurred when creating an IUCV_port and socket connection.
(tcperror) is the number of the message where you can find more detailed information about this error.
System action
The application continues.

Operator response
Notify the system programmer of the error.

System programmer response
Respond as indicated in \( tcperror \). For more information on the error condition see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
MISCSRV

Procedure name
init_SMSGsocket

EZY2430E cannot connect smsg socket \( tcperror \)

Explanation
An error was detected when trying to initiate a socket connection to the server. No connection is established. \( tcperror \) is the number of the message where you can find more detailed information about this error.

System action
The application continues.

Operator response
Notify the system programmer of the error.

System programmer response
Respond as indicated in \( tcperror \). A description of the connect() function and error codes for the function can be found in the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
MISCSRV

Procedure name
init_SMSGsocket

EZY2431E Accept() \( tcperror \)

Explanation
The accept() call, which is used to establish a connection to a client, failed to respond to a connection request. \( tcperror \) is the number of the message where you can find more detailed information about this error.

System action
The application continues.
**Operator response**

Notify the system programmer of the error.

**System programmer response**

Respond as indicated in `(tcperror)`. A description of the `recv()` call and the error conditions generated from this call can be found in the *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference*.

**Module**

MISCSRV

**Procedure name**

doTCPecho

**EZY2432E recv() (tcperror)**

**Explanation**

The `recv()` function, which is used to receive data from a socket and store it in a buffer, was unsuccessful. `(tcperror)` is the number of the message where you can find more detailed information about this error.

**System action**

The MISCSRV halts. The application continues.

**Operator response**

Notify the system programmer of the error.

**System programmer response**

Respond as indicated in `(tcperror)`. For more information about the `recv()` call and the error conditions for this call, see the *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference*.

**Module**

MISCSRV

**Procedure name**

doTCPecho

**EZY2433E error reading smsg_socket (tcperror)**

**Explanation**

The buffer space used to store data read from the `smsg_socket` cannot accommodate the data. The following errors can occur as a result of the failure:

**EBADF**

The socket is not a valid socket descriptor.

**EFAULT**

Exceeded storage for the caller address space.

**EWOULDBLOCK**

The socket is in nonblocking mode and data is not available to read.

`(tcperror)` is the number of the message where you can find more detailed information about this error.
**System action**
The application continues.

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

**System programmer response**
Respond as indicated in *(tcperror)*. For more information see the *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference*.

**Module**
MISCSRV

**Procedure name**
dispSMSG

**EZY2434E**  create_UDPC_port: ioctl (get interface conf.) *(tcperror)*

**Explanation**
The ioctl function, which is used to get the network configuration, encountered an error. For more information about the ioctl function, see the *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference*.

*(tcperror)* is the number of the message where you can find more detailed information about this error.

**System action**
The MISCSRV server halts. The application continues.

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

**System programmer response**
Respond as indicated in *(tcperror)*.

**Module**
MISCSRV

**Procedure name**
getifconf

**EZY2435I**  Echo server: socket number buf value

**Explanation**
This message displays the server type, socket number, and contents of the buffer that holding data to be transmitted.

**System action**
The application continues.
**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
doecho

**EZY2436I** Discard server: socket number buf value

**Explanation**
This message displays the server type, socket number, and contents of the buffer holding received data.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
dodisc

**EZY2437I** Character server: socket number buf value

**Explanation**
This message displays the server type, socket number, and the data stored in the buffer.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.
Module
MISCSRV

Procedure name
dochargen

EZY2438I Invalid Trace option name

Explanation
The trace option requested is not valid. The following are valid trace options:

(NO)ECHO
Returns the data for UDP and TCP sessions exactly as it is received.

(NO)DISC
The data is discarded when sent to the MISCRSV.

(NO)CHARGEN
Random data is displayed regardless of the data it receives.

System action
The application continues.

Operator response
Reenter the trace option with a valid request.

System programmer response
Assist the user as necessary.

Module
MISCSRV

Procedure name
dispSMSG

EZY2439I Currently not tracing anything.

Explanation
The trace option is not enabled.

System action
The application continues.

Operator response
None.

System programmer response
None.

Module
MISCSRV
**Procedure name**
dispSMSG

**EZY2440I** Currently tracing: name

**Explanation**
This informational message displays the current level of tracing.

**System action**
The application continues processing.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
dispSMSG

**EZY2441I** Trace option not defined

**Explanation**
The application does not support the trace option.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MISCSRV

**Procedure name**
main

**EZY2442I** Invalid initialization parameter: name

**Explanation**
The parameter you have entered is not valid.
**System action**
The application continues.

**Operator response**
Restart the miscellaneous server with a valid parameter. See the z/OS Communications Server: IP User's Guide and Commands for more information about correct parameters.

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

**Module**
MISCSRV

**Procedure name**
main

**EZY2443I**  
fnctl failed: *(tcperror)*

**Explanation**
The fnctl() call failed to initialize the socket descriptor into nonblocking mode. *(tcperror)* is the number of the message where you can find more detailed information about this error.

**System action**
Initialization of the MISCSRV server halts. The application continues.

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

**System programmer response**
Respond as indicated in *(tcperror)*. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information about the fnctl() call and the error conditions that can occur when using this call.

**Module**
MISCSRV

**Procedure name**
init_TCPsocket

**EZY2632E**  
*parameter* value must be either TRUE or FALSE.

**Explanation**
While processing the FTP.DATA file, a parameter in the file was encountered which required a value of either TRUE or FALSE. The actual value specified was something other than TRUE or FALSE. *parameter* is the name of the parameter in error.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.
**Operator response**
Contact the System programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to contain the correct value for parameter. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

**EZY2633E  parameter value missing - must be either TRUE or FALSE.**

**Explanation**
While processing the FTP.DATA file, a parameter in the file was encountered which required a value of either TRUE or FALSE, but which had no value specified. Parameter is the name of the parameter in error.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

**Operator response**
Contact the System programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

**EZY2634E  parameter value value not numeric.**

**Explanation**
While processing either the FTP.DATA file or processing the start options from the FTP server start procedure, a parameter was encountered which required a numeric value, but which had a non-numeric value specified. parameter is the name of the parameter in error. value is the value which was found in the FTP.DATA file or on the start options for the parameter. This message will be preceded by either message EZYFT46E or EZY2655E indicating whether the error was in the start procedure or in the FTP.DATA file.

**System action**
The parameter in error is ignored. Processing of the FTP.DATA file continues with the next line in the file. Processing of the FTP server procedure start options continues with the next start option.

**Operator response**
Contact the System programmer with the error message to have the FTP.DATA file or FTP server start procedure corrected.

**System programmer response**
Correct the FTP.DATA file or the FTP server procedure to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file and FTP server start options.
EZY2635E  Integer overflow number for parameter

Explanation
While processing either the FTP.DATA file or the FTP server procedure start options, a parameter was encountered which required a numeric value, but the value which was specified was larger than the largest valid 4 byte integer value. parameter is the name of the parameter in error. number is the value which was found in the FTP.DATA file or FTP server procedure start options for the parameter. This message will be preceded by either message EZYFT46E or EZY2655E indicating whether the error was in the start procedure or in the FTP.DATA file.

System action
The parameter in error is ignored. Processing of the FTP.DATA file continues with the next line in the file. Processing of the FTP server start option continues with the next start option.

Operator response
Contact the System programmer with the error message to have the FTP.DATA file or FTP server start procedure corrected.

System programmer response
Correct the FTP.DATA file or FTP server start procedure to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file and FTP server start options.

EZY2636E  parameter value not specified.

Explanation
While processing the FTP.DATA file, a parameter in the file was encountered which required a value, but no value was specified. parameter is the name of the parameter in error.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

Operator response
Contact the System programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Tip: If the configuration input is from a UNIX file (see the preceding EZYFT46E message), this might be caused by trailing blanks in the configuration statement. Edit the file to remove those blanks or add a comment to the end of the statement.
Using FTP configuration defaults.

Explanation
This message is issued by the FTP daemon or by the FTP client.

FTP daemon
The FTP daemon could not locate or open an FTP.DATA file or data set when the FTP daemon was started. The FTP daemon logs this message to SYSLOG during initialization after it attempts to locate and process FTP.DATA.

FTP client
The FTP client could not locate or open an FTP.DATA file or data set when the FTP client was started. The FTP client issues this message as part of the LOCSTat subcommand response.

The internal defaults for all FTP.DATA configuration options are being used.

See the z/OS Communications Server: IP Configuration Reference about FTP.DATA configuration options.

System action
The configurable parameters of the FTP client or server are set to default values.

Operator response
No action needed.

System programmer response
See the z/OS Communications Server: IP Configuration Reference for information about configuration options for the FTP.DATA file or data set.

If you want to customize the client or server configuration, perform one of the following actions:

FTP client
• Create an FTP.DATA file or data set. See the Changing local site defaults using FTP.DATA in z/OS Communications Server: IP User's Guide and Commands for information about creating an FTP.DATA file or data set. If an FTP.DATA file or data set already exists, error messages issued when the FTP client was started indicate why the existing FTP.DATA file or data set was not used.
• Issue the LOCSIte subcommand to change the configuration for the current session only. See the LOCSIte subcommand in z/OS Communications Server: IP User's Guide and Commands for more information.

FTP daemon
• Create an FTP.DATA file or data set. See the Configuring FTP.DATA in z/OS Communications Server: IP Configuration Guide for information about creating FTP.DATA. If an FTP.DATA file or data set already exists, check SYSLOGD for error messages from the FTP daemon indicating why the existing FTP.DATA file or data set was not used.
• Send a SITE command to the server after logging in to change the configuration for the current session only. See the SIte subcommand in z/OS Communications Server: IP User's Guide and Commands for information about configuring the FTP server with the SIte subcommand and SITE command.
• Issue a MODIFY command from the console to change the configuration for subsequent sessions only. The modified configuration is lost when you stop the FTP daemon. See the MODIFY command: FTP in z/OS Communications Server: IP System Administrator’s Commands for information about the MODIFY command.
**Explanation**
This message indicates which FTP.DATA file is being used for the configurable parameters. *name* will either be "DD:SYSFTPD", indicating that the SYSFTP DD statement was used for the FTP.DATA file, or it will be the actual name of the file being used. See the z/OS Communications Server: IP Configuration Reference for information about the FTP.DATA file search order.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPEP

---

**EZY2642E** Unknown keyword: *parameter*

**Explanation**
While processing the FTP.DATA file, the FTP server encountered a parameter which it did not recognize.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

**Operator response**
Contact the System programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Check the line in error for spelling or typographical errors. If the parameter is incorrect, correct the parameter as necessary. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

---

**EZY2643E** *parameter* value longer than *number* characters.

**Explanation**
While processing the FTP.DATA file, a parameter in the file was encountered which required a character string no longer than the specified number of characters. However, the value that was specified was longer than the maximum allowed length. *parameter* is the parameter in error. *number* is the maximum allowed length of the parameter value.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.
Operator response
Contact the System programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPEP

EZY2644E parameter value number out of range. Value must be between low and high.

Explanation
While processing either the FTP.DATA file or processing the start options from the FTP server start procedure, a parameter was encountered which required a numeric value which was within a specified range. The actual value specified was outside of the allowable range of values for the parameter being processed. parameter is the parameter in error. number is the actual value that was specified. low is the minimum acceptable value, high is the maximum allowable value. This message will be preceded by either message EZYFT46E or EZY2655E indicating whether the error was in the start procedure or in the FTP.DATA file.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file. Processing of the FTP server procedure start parameters continues with the next start parameter.

Operator response
Contact the System programmer with the error message to have the FTP.DATA file or FTP server start procedure corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPEP

EZY2645E CONDDISP value must be either CATLG or DELETE.

Explanation
While processing the FTP.DATA file, the server encountered the CONDDISP parameter with a parameter value that was not CATLG or DELETE. The only valid values for the CONDDISP parameter are CATLG or DELETE.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

Operator response
Contact the System programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.
Module
EZAFTPEP

**EZY2648E**  FILETYPE value must be either SEQ, JES, or SQL.

**Explanation**
While processing the FTP.DATA file, the FTP server encountered the FILETYPE parameter, but the value specified for the FILETYPE parameter was not SEQ, JES, or SQL.

**System action**
The line containing the FILETYPE parameter is ignored. Processing continues with the next line in the file.

**Operator response**
Contact the system programmer with the error message.

**System programmer response**
Correct the value of the FILETYPE parameter in the FTP.DATA file to be either SEQ, JES, or SQL. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPEP

**EZY2651E**  UNITNAME parameter unit: invalid unit type.

**Explanation**
While processing the FTP.DATA file, the FTP server encountered the UNITNAME parameter, but the value specified for the UNITNAME parameter was not a valid DASD or TAPE unit as defined by the host MVS system.

**System action**
The line containing the UNITNAME parameter is ignored. Processing continues with the next line in the file.

**Operator response**
Contact the system programmer with the error message.

**System programmer response**
Correct the value of the UNITNAME parameter in the FTP.DATA file to be a valid DASD or TAPE unit. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPEP

**EZY2655E**  Error in start parameters.

**Explanation**
While processing the start options specified in the FTP server start procedure, the FTP Server encountered an error.

**System action**
The parameter in error is ignored. Processing of the FTP server start options continues with the next start option.
Operator response
This message should be followed by another message which describes the error. Contact the System programmer with both messages to have the FTP server start procedure corrected.

System programmer response
This message should be followed by another message which describes the error. Correct the error described by the second message.

Module
EZAFTPEP

EZY2657W Host name unknown; gethostname() error: error

Explanation
The gethostname() socket call did not complete successfully.

System action
Processing continues. The hostname will not be known to the server.

Operator response
Correct the error indicated by the error.

System programmer response
Correct the error indicated by the error

Module
EZAFTPDM

EZY2658W domain name unknown: gethostbyname() error.

Explanation
The gethostbyname() socket call did not complete successfully.

System action
Processing continues. The host domain will not be known to the server.

Operator response
Contact the system programmer.

System programmer response
Ensure that the host name is accessible via a name server, or defined in HOSTS.SITEINFO and HOSTS.ADDRINFO. See the z/OS Communications Server: IP Configuration Reference for information on domain name systems or HOSTS.SITEINFO and HOSTS.ADDRINFO data sets.

Module
EZAFTPDM

EZY2659E RECFM value must be one of: F, FA, FB, FBM, FBS, FBSA, FBSM, FM, FS, FSA, FSM, V, VA, VB, VBA, VBM, VBS, VBSA, VBSM, VM, VS, VSA, VSM, U, UA, or UM.
Explanation
While processing the FTP.DATA file, the FTP server encountered the RECFM parameter, but the value specified for the parameter was not a valid value. The value must be one of the values listed in the message.

System action
The line containing the RECFM parameter is ignored. Processing continues with the next line in the file.

Operator response
Contact the system programmer with the error message.

System programmer response
Correct the value of the RECFM parameter in the FTP.DATA file to be a valid record format. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPEP

EZY2669E  SPACETYPE value must be either BLOCK, TRACK, or CYLINDER.

Explanation
While processing the FTP.DATA file, the FTP server encountered the SPACETYPE parameter, but the value specified for the parameter was not a valid value. The value must be one of the values listed in the message.

System action
The line containing the SPACETYPE parameter is ignored. Processing continues with the next line in the file.

Operator response
Contact the system programmer with the error message.

System programmer response
Correct the value of the SPACETYPE parameter in the FTP.DATA file to be a valid space type. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPEP

EZY2670E  SQLCOL value must be either NAMES, LABELS, or ANY.

Explanation
While processing the FTP.DATA file, the FTP server encountered the SQLCOL parameter, but the value specified for the parameter was not a valid value. The value must be one of the values listed in the message.

System action
The line containing the SQLCOL parameter is ignored. Processing continues with the next line in the file.

Operator response
Contact the system programmer with the error message.
**System programmer response**
Correct the value of the SQLCOL parameter in the FTP.DATA file to be a valid column heading. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

**EZY2671I** General tracing is already active.

**Explanation**
A MODIFY command with the TRACE option has been issued while tracing of FTP's general activity is currently active. General tracing continues.

**System action**
FTP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPHD

**EZY2672E** read_ftpdata invalid keycode encountered: key_code

**Explanation**
While processing the FTP.DATA file, an internal error occurred.
key_code is an internal code describing the error.

**System action**
An assertion failure occurs to provide additional information about the error.

**Operator response**
Contact the system programmer with the error message and the assertion failure output.

**System programmer response**
Contact the IBM Software Support Center with the error and the assertion failure output.

**Module**
EZAFTPEP

**EZY2673E** Error reading FTP configuration file: error_description

**Explanation**
While reading the records in the FTP.DATA set, the read of one of the records failed.
error_description describes the error.
System action
Processing of the FTP.DATA file is terminated.

Operator response
Correct the problem with the FTP.DATA file.

System programmer response
Correct the problem with the FTP.DATA file.

Module
EZAFTPEP

**EZ2674E**  LRECL *number* is invalid for RECFM *recfm*.

Explanation
After processing the FTP.DATA file, the server cross checked the new values of the RECFM and LRECL variables and found them to be set to a combination that is invalid for the MVS operating system.

System action
The LRECL value is changed to the nearest value which is compatible with the record format. This message should be followed by EZ2675I, indicating the new value of the LRECL.

Operator response
If the new value is acceptable, no action is necessary, otherwise contact the system programmer to get the correct lrecl and record format values put into the FTP.DATA file.

System programmer response
Update the FTP.DATA file with the corrected values for RECFM or LRECL. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPEP

**EZ2675I**  LRECL being reset to *number*

Explanation
After processing the FTP.DATA file, the server cross checked the new values of the RECFM and LRECL parameters and found them to be set to a combination that is invalid for the MVS operating system. The LRECL parameter was changed to the indicated lrecl. This message is usually preceded by message EZ2674E indicating the error.

System action
LRECL parameter is changed.

Operator response
Correct the error as indicated for message EZ2674E.

System programmer response
Correct the error as indicated for message EZ2674E.
**EZY2676E**  BLOCKSIZE must equal LRECL for RECFM *recfm*.

**Explanation**
After processing the FTP.DATA file, the server cross checked the new values of the RECFM, LRECL, and BLOCKSIZE parameters and found the RECFM parameter set to a record format which requires the LRECL and BLOCKSIZE parameters to be equal, but they were not equal. The BLOCKSIZE parameter was changed to the indicated lrecl. This message is followed by message EZY2677I indicating the new value of BLOCKSIZE.

**System action**
The blocksize value is changed to the lrecl value.

**Operator response**
If the new value is acceptable, no action is necessary, otherwise contact the system programmer to get the correct lrecl, blocksize, and record format values put into the FTP.DATA file.

**System programmer response**
Update the FTP.DATA file with the corrected values for RECFM, LRECL, or BLOCKSIZE or all three. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

**EZY2677I**  BLOCKSIZE being set to blocksize number.

**Explanation**
After processing the FTP.DATA file, the server cross checked the new values of the RECFM, LRECL, and BLOCKSIZE parameters and found the RECFM parameter set to a record format which requires the value of the BLOCKSIZE parameter to have a dependency on the value of the LRECL parameter, but the dependency was not met. The BLOCKSIZE parameter was changed to comply with the dependency on the LRECL. This message is preceded by message EZY2676E, EZY2678E, or EZY2679E indicating the nature of the relationship between BLOCKSIZE and LRECL.

**System action**
The blocksize value is changed to the indicated value.

**Operator response**
Correct the error as indicated for message EZY2676E, EZY2678E, or EZY2679E.

**System programmer response**
Correct the error as indicated for message EZY2676E, EZY2678E, or EZY2679E.

**Module**
EZAFTPEP

**EZY2678E**  BLOCKSIZE must be a multiple of LRECL for RECFM *recfm*.
**Explanation**
After processing the FTP.DATA file, the server cross checked the new values of the RECFM, LRECL, and BLOCKSIZE parameters and found the RECFM parameter set to a record format which requires the BLOCKSIZE to be a multiple of the LRECL parameter, but it was not.

**System action**
BLOCKSIZE is set to a value that is the nearest multiple of LRECL.

**Operator response**
If the new value is acceptable, no action is necessary, otherwise contact the system programmer to get the correct lrecl, blocksize, and record format values put into the FTP.DATA file.

**System programmer response**
Update the FTP.DATA file with the corrected values for RECFM, LRECL, or BLOCKSIZE or all three. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

---

**EZY2679E** BLOCKSIZE must be at least 4 more than LRECL for RECFM 

**Explanation**
After processing the FTP.DATA file, the server cross checked the new values of the RECFM, LRECL, and BLOCKSIZE parameters and found the RECFM parameter set to a record format which requires the BLOCKSIZE to be at least four greater than the LRECL parameter, but it was not.

**System action**
BLOCKSIZE is set to LRECL + 4. If LRECL is greater than 32756, the LRECL will be changed to 32756 so that BLOCKSIZE will not exceed the maximum value of 32760.

**Operator response**
If the new value is acceptable, no action is necessary, otherwise contact the system programmer to get the correct lrecl, blocksize, and record format values put into the FTP.DATA file.

**System programmer response**
Update the FTP.DATA file with the corrected values for RECFM, LRECL, or BLOCKSIZE or all three. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

---

**EZY2680I** LRECL being changed to number

**Explanation**
After processing the FTP.DATA file, the server cross checked the new values of the RECFM, LRECL, and BLOCKSIZE parameters and found the RECFM parameter set to a record format which requires the BLOCKSIZE to be at least four greater than the LRECL parameter, but it was not. The server attempted to change the BLOCKSIZE value to LRECL + 4, but the LRECL value was greater than 32756, which would cause the BLOCKSIZE to exceed the maximum value of 32760. LRECL was changed to the indicated value.
System action
LRECL is changed to the indicated value.

Operator response
Correct the error as indicated for message EZY2679E.

System programmer response
Correct the error as indicated for message EZY2679E.

Module
EZAFTPEP

**EZY2681E** Invalid record format 'recfm' encountered.

Explanation
After processing the FTP.DATA file, the server attempted to cross check the RECFM, BLOCKSIZE, and LRECL values, but the value of RECFM was invalid.

System action
The RECFM, BLOCKSIZE, and LRECL values are reset to the default values.

Operator response
Contact the system programmer with the error message.

System programmer response
Correct the FTP.DATA statement by choosing the valid values.

Module
EZAFTPEP

**EZY2682I** LRECL, RECFM, and BLOCKSIZE being reset to default values.

Explanation
After processing the FTP.DATA file or LOCSITE command, the server or client attempted to cross check the RECFM, BLOCKSIZE, and LRECL values, but the value of RECFM was invalid. The LRECL, RECFM, and BLOCKSIZE values are reset to the default values.

System action
The RECFM, BLOCKSIZE, and LRECL values are reset to the default values.

Operator response
Contact the system programmer if you do not want to use the default values.

System programmer response
Correct the FTP.DATA statement by choosing the valid values.

Module
EZAFTPEP
EZY2688E  Unknown start option: option

Explanation
While processing the start options specified in the FTP server start procedure, the FTP server encountered the invalid start option listed in the message.

System action
The start option is ignored. Processing continues with the next start option.

Operator response
Contact the system programmer to correct the FTP server start procedure.

System programmer response
Correct the start option in the FTP server start procedure. See the z/OS Communications Server: IP Configuration Reference for information on the FTP Server start options.

Module
EZAFTPEP

EZY2689E  An error occurred processing the FTP translate table file name.

Explanation
FTP encountered an error while attempting to process the translation table file name.

System action
The FTP server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the internally provided translation tables are used.

Operator response
This message is followed by an additional message that further defines the error. Correct the error indicated by these messages.

System programmer response
This message is followed by an additional message that further defines the error. Correct the error indicated by these messages.

Module
EZAFTPNY

EZY2690E  Header record invalid format.

Explanation
The FTP server was attempting to load a translation table, but the file specified for the translation table did not have a valid TCPXLBIN header record. The first record in the file should have been "*TCP/IP translate tables", but was not.

System action
The FTP server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the internally provided translation tables are used.
Operator response
If the file name is correct, the file has probably gotten corrupted. Use the CONVXLATE command to rebuild the table.

System programmer response
If the file name is correct, the file has probably gotten corrupted. Use the CONVXLATE command to rebuild the table.

Module
EZAFTPNY

EZY2691E  Error reading the file: error

Explanation
The FTP server was attempting to load a translation table, but the error error occurred while attempting to read the file.

System action
The FTP server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the internally provided translation tables are used.

Operator response
Correct the error indicated by error. It might be necessary to rebuild the TCPXLBIN file using the CONVXLATE command.

System programmer response
Correct the error indicated by error. It might be necessary to rebuild the TCPXLBIN file using the CONVXLATE command.

Module
EZAFTPNY

EZY2693I  Unable to open name : reason

Explanation
The FTP server attempted to open the specified file, but the open failed for the specified reason.

System action
The FTP server will continue through the search order.

Operator response
If the file name is the wanted file, correct the error specified by reason. This message will be issued for each file as the server proceeds through the search order. If the wanted file is further in the search order, no action is necessary.

System programmer response
If the file name is the wanted file, correct the error specified by reason. This message will be issued for each file as the server proceeds through the search order. If the wanted file is further in the search order, no action is necessary.
Module
EZAFTPCZ, EZAFTPEP, EZAFTPNX

EZY2697I  IBM FTP version time on date

Explanation
This is an informational message indicating that the MVS FTP server has completed initialization and is ready for client connections. The message indicates the version of the FTP server and the time and date the server was started.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPDM

EZY2700I  Using port FTP control (portnumber)

Explanation
This is an informational message indicating which port the FTP server is listening on for incoming client connections. This should be either the port specified for the ftp server in the etc.services file, the port specified by the PORT start option, or default port 21.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPDM

EZY2701I  Inactivity time is timeout

Explanation
This is an informational message indicating the value being used for the client control connection timeout. Client control connections inactive for this number of seconds will be terminated by the server. A timeout value of 0 indicates that inactive client connections will not be terminated. This value should be either the default timeout value or the value specified by the INACTIVE start option or FTP.DATA parameter.
**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPD

**EZY2702I**  Server-FTP: Initialization completed at time on date

**Explanation**
This is an informational message indicating that the MVS FTP C Server has completed initialization and is ready for client connections. The message indicates the time and date the server was started.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPSD

**EZY2706E**  Unsupported Modify command parameter parameter ignored.

**Explanation**
A Modify request was issued to the Server, but the server did not recognize the parameter entered on the modify request.

**System action**
The request is ignored.

**Operator response**
Verify that the Modify command was entered correctly. See the z/OS Communications Server: IP Configuration Reference for information on the Modify command.

**System programmer response**
None.

**Module**
EZAFTPDF
**EZY2707E**  Unsupported console command (hex code) *command name* ignored.

**Explanation**
An MVS command was entered to the FTP server, but the server did not support the MVS command. Currently the only supported MVS commands are MODIFY and STOP.

**System action**
The MVS command is ignored.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPDF

---

**EZY2714I**  FTP server shutdown in progress

**Explanation**
The FTP daemon has been terminated either by an MVS operator STOP command, or by an OMVS kill command.

**System action**
The FTP daemon terminates.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPDH

---

**EZY2715I**  General tracing is not active.

**Explanation**
A MODIFY command was entered requesting that general tracing of all user IDs be ended, but tracing is not currently active. The MODIFY command is ignored.

**System action**
FTP continues.

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

Module
EZAFTPHD

EZY2720I Using Japanese translation tables in filename

Explanation
The translation table in the file is used as specified.

System action
FTP continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPDC

EZY2721E Unable to load Japanese translation tables in filename; return code: code

Explanation
The FTP server encountered an error while attempting to load the indicated translation table. Possible return codes are:

1
   No storage for the translation table

2
   Unable to find the translation table header

3
   Error reading the file

System action
FTP continues.

Operator response
Notify the system programmer of the error.

System programmer response
Respond as indicated by the return code displayed in the message.

Module
EZAFTPDC

EZY2722E Unable to load Japanese translation tables.
**Explanation**
The FTP server cannot open the indicated translation table.

**System action**
FTP continues.

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

**System programmer response**
Make sure that the translation table is properly specified and loaded in storage accessible to the FTP server.

**Module**
EZAFTPDC

**EZY2723I Using Korean translation tables in filename**

**Explanation**
The FTP server is using the indicated translation table.

**System action**
FTP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPDC

**EZY2724E Unable to load Korean translation tables in filename; return code: code**

**Explanation**
The FTP server was unable to load the indicated translation table. Possible return codes are:

1
   No storage for the translation table

2
   Unable to find the translation table header

3
   Error reading the file

**System action**
FTP continues.

**Operator response**
Notify the system programmer of the error.
**System programmer response**
Respond as indicated by the return code displayed in this message.

**System programmer response**
None.

**Module**
EZAFTPDC

**EZY2725E**  Unable to load Korean translation tables.

**Explanation**
The FTP server cannot open the indicated translation table.

**System action**
FTP continues.

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

**System programmer response**
Make sure that the translation tables are properly specified and in storage accessible to the FTP server.

**Module**
EZAFTPDC

**EZY2726I**  Using traditional Chinese translation tables in `filename`

**Explanation**
The FTP server is using the indicated Chinese translation table.

**System action**
FTP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPDC

**EZY2727E**  Unable to load traditional Chinese translation tables in `filename`; return code: `code`

**Explanation**
The FTP server cannot open the indicated Chinese translation table. Possible return codes are:

342  z/OS Communications Server: IP Messages: Volume 3 (EZY)
No storage for the translation table

Unable to find the translation table header

Error reading the file

**System action**
FTP continues.

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

**System programmer response**
Respond as indicated by the return code displayed in this message.

**Module**
EZAFTPDC

**EZY2728E** Unable to load traditional Chinese translation tables.

**Explanation**
The FTP server cannot open the indicated Chinese translation table.

**System action**
FTP continues.

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

**System programmer response**
Make sure that the Chinese translation table is properly specified and in storage accessible to the FTP server.

**Module**
EZAFTPDC

**EZY2790E** Invalid format data set name "dsn". Dcbdsn parameter ignored.

**Explanation**
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, dsn, was not a valid MVS data set name. The DCBDSN keyword is ignored. The FTP server continues with the next keyword.

**System action**
FTP continues.

**Operator response**
Notify the system programmer of the error.
System programmer response
Correct the DCBDSN data set name in the FTP.DATA file. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPRU

**EZY2792E** Error retrieving "data_set". Dcbdsn parameter ignored.

Explanation
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, data_set, was migrated, and the server was unable to recall the data set. The keyword is ignored. The FTP server continues with the next keyword.

System action
FTP continues.

Operator response
Determine why the data set cannot be recalled and correct the error.

System programmer response
Assist the user as necessary.

Module
EZAFTPRU

**EZY2793E** "data_set" is migrated and noautorecall is specified. Dcbdsn parameter ignored.

Explanation
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, data_set, was migrated and No Autorecall had also been specified. The server was unable to recall the data set. The keyword is ignored. The FTP server continues with the next keyword.

System action
FTP continues.

Operator response
Notify the system programmer of the error.

System programmer response
Change the FTP.DATA file or the FTP server start procedure to specify Autorecall; or recall the data set prior to starting the FTP server (note that the data set must then not be migrated at any point while the FTP server is active); or change the data set specified for the DCBDSN keyword to a data set that is not migrated. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPRU

**EZY2794E** Error mounting "data_set". Dcbdsn parameter ignored.
**Explanation**
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, *data_set*, was on an unmounted volume. The server encountered an error attempting to mount the volume. The keyword is ignored and the FTP server continues processing with the next keyword.

**System action**
FTP continues.

**Operator response**
Determine why the volume could not be mounted and correct the error.

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

**Module**
EZAFTPRU

**EZY2795E** Volume for "*data_set*" is not mounted and noautomount is specified. DCBDSN parameter ignored.

**Explanation**
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, *data_set*, was on an unmounted volume and NO AUTOMOUNT had also been specified. The server was unable to mount the volume. The keyword is ignored and the FTP server continues with the next keyword.

**System action**
FTP continues.

**Operator response**
Contact the system programmer to change the settings of the FTP.DATA file. If necessary, mount the volume containing the DCBDSN data set.

**System programmer response**
Change the FTP.DATA file or server job start options to allow Automount; mount the volume containing the DCBDSN data set (note that the volume must then remain mounted for the duration of the FTP server job); or change the DCBDSN parameter to specify a data set that is on a mounted volume. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPRU

**EZY2796E** Data set "*data_set*" does not exist. Dcbdsn parameter ignored.

**Explanation**
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, *data_set*, did not exist. The keyword is ignored and the FTP server continues with the next keyword.
**System action**
FTP continues.

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

**System programmer response**
Correct the FTP.DATA file to specify an existing data set for the DCBDSN keyword. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPRU

**EZY2797E** "data_set" is not on a direct access volume. Dcbdsn parameter ignored.

**Explanation**
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, data_set, was not located on a direct access volume. The keyword is ignored and the FTP server continues with the next keyword.

**System action**
FTP continues.

**Operator response**
Notify the system programmer.

**System programmer response**
Change the DCBDSN parameter of the FTP.DATA file to specify a data set that is on a direct access volume. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

**Module**
EZAFTPRU

**EZY2798E** "data_set" is a VSAM data set. Dcbdsn parameter ignored.

**Explanation**
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, data_set, was a VSAM data set and could not be used as a model data set. The keyword is ignored and the FTP server continues with the next keyword.

**System action**
FTP continues.

**Operator response**
Contact the system programmer to change the settings of the FTP.DATA file.
System programmer response
Change the DCBDSN keyword in the FTP.DATA file to specify a non-VSAM data set. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPRU

**EZY2799E** "data_set" invalid dsorg. Dcbdsn parameter ignored.

Explanation
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, data_set, had an incorrect data set organization and could not be used as a model data set. The keyword is ignored and the FTP server continues with the next keyword.

System action
FTP continues.

Operator response
Contact the system programmer to change the settings of the FTP.DATA file.

System programmer response
Change the DCBDSN keyword in the FTP.DATA file to specify a data set with a valid data set organization (PS or PO). See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPRU

**EZY2800E** Error locating file "data_set". Dcbdsn parameter ignored.

Explanation
The FTP server was processing the DCBDSN keyword in the FTP.DATA file. The data set name specified for the DCBDSN data set, data_set, could not be located. The keyword is ignored and the FTP server continues with the next keyword.

System action
FTP continues.

Operator response
If the data set exists, determine why the server was unable to locate it, and correct the error. If necessary, contact the system programmer to change the settings of the FTP.DATA file.

System programmer response
Change the DCBDSN keyword in the FTP.DATA file to specify a valid data set. See the z/OS Communications Server: IP Configuration Reference for information on the parameters of the FTP.DATA file.

Module
EZAFTPRU

**EZY2830I** Using simplified Chinese translation tables in *filename*
Explanation
The FTP server is using the indicated simplified Chinese translation table.

System action
FTP continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPDC

EZY2831E Unable to load simplified Chinese translation tables in filename; return code: code

Explanation
The FTP server cannot open the indicated simplified Chinese translation table. Possible return codes are:

1  No storage for the translation table
2  Unable to find the translation table header
3  Error reading the file

System action
FTP continues.

Operator response
Notify the system programmer of the error.

System programmer response
Respond as indicated by the return code displayed in this message.

Module
EZAFTPDC

EZY2832E Unable to load simplified Chinese translation tables.

Explanation
The FTP server cannot open the indicated simplified Chinese translation table.

System action
FTP continues.

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

Make sure that the simplified Chinese translation table is properly specified and in storage accessible to the FTP server.

**Module**

EZAFTPDC
Chapter 5. EZY3xxxx messages

EZY3720I  

Explaination
This message is part of the socket trace used for problem analysis and can be generated by specifying trace socket in the TCPIP.DATA configuration file.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
print_request_parms

EZY3721I

Explaination
This message is part of the socket trace used for problem analysis and can be generated by specifying trace socket in the TCPIP.DATA configuration file.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
print_request_parms

EZY3722I  

string hexvalue xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx
Explanation
Hexadecimal display of information starting at storage address hexvalue. This display is generated by a call to the prntdata function. The string value is supplied by the caller to identify this output. If more than 16 bytes of data are displayed, message EZY3723I is used for additional lines.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
prntdata

EZY3723I string xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx

Explanation
Hexadecimal storage display generated by a call to the prntdata function. This format is used for subsequent lines after message EZY3722I until the requested amount of data is displayed.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
prntdata

EZY3724I descarray is at addr, size is bytes bytes

Explanation
This indicates the address and size of the socket descriptor array.
This message is part of the socket trace used for problem analysis and can be generated by specifying trace socket in the TCPIP.DATA configuration file.
System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
alloc_global_fdsets

EZY3725I  descarray has number entries, entry size is bytes

Explanation
This indicates the number of socket descriptor entries and the size of the entries that appear in the IUCV socket descriptor table.

This message is part of the socket trace used for problem analysis and can be generated by specifying trace socket in the TCPIP.DATA configuration file.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
alloc_global_fdsets

EZY3726I  iucvdesc is at addr

Explanation
The IUCV descriptor structure is located at the address in this message. This is the location of extra storage for IUCV socket information.

This message is part of the socket trace used for problem analysis and can be generated by specifying trace socket in the TCPIP.DATA configuration file.

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

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
alloc_global_fdsets

---

**EZY3727I** in nextbuf with function type

**Explanation**
The function descarray_nextbuf has been called with a type argument other than receive, send, read, accept, or write. The function descarray_nextbuf issues this message and returns -1.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
descarray_nextbuf

---

**EZY3728I** data pending=*

**Explanation**
This indicates the pointers to the number of input buffers left to be read, the file descriptor, the IUCV path ID, the IUCV parameter list status type, and the actual status.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace socket` in the TCPIP.DATA configuration file.

**System action**
TCPIP continues.

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

**Module**  
CMIUCSOC

**Procedure name**  
descarray_iucv_receive

**EZY3729I**  
fd=descriptor iucvselect now TRUE

**Explanation**  
This indicates the file descriptor and that at least one AF_IUCV descriptor is in the caller's mask.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace socket` in the TCPIP.DATA configuration file.

**System action**  
TCP/IP continues.

**Operator response**  
None.

**System programmer response**  
None.

---

**Module**  
CMIUCSOC

**Procedure name**  
descarray_select

**EZY3730I**  
fd=descriptor inetselect now TRUE

**Explanation**  
This indicates the file descriptor and that at least one AF_INET descriptor is in the caller's mask.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace socket` in the TCPIP.DATA configuration file.

**System action**  
TCP/IP continues.

**Operator response**  
None.

**System programmer response**  
None.
**Explanation**

This message is issued when debugging is requested (a trace socket statement appears in the tcpip.data configuration file). It is issued along with message EZY3732X to indicate that an IUCV interrupt has been received from TCPIP.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

None.

---

**Explanation**

This message is concatenated to EZY3731I and indicates the file descriptor, the path ID, the numeric value of the interrupt type, and the symbolic type of external interrupt.

**System action**

TCPIP continues

**Operator response**

None.

**System programmer response**

None.
**EZY3733E**  
userid severed IUCV path.

**Explanation**
The indicated user severed the IUCV path. The TCP connection corresponding to each stream socket is reset.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
descarray_save_iucvbuf

---

**EZY3734E**  
userid severed IUCV path. Reason .ipuser.field

**Explanation**
The indicated userid severed the IUCV path. This occurs only in the case of shutdown or an unexpected error. The ipuser field in the SEVER external interrupt indicates the reason for the sever. The reason is in EBCDIC. The possible reason codes and explanations appear in z/OS Communications Server: IP Programmer's Guide and Reference.

**System action**
The system action depends on the type of error that occurred.

**Operator response**
None.

**System programmer response**
See z/OS Communications Server: IP Programmer's Guide and Reference to determine the reason for this error and to correct the error. If the error is the result of an error in TCPIP, contact the IBM Software Support Center.

**Module**
CMIUCSOC

**Procedure name**
descarray_save_iucvbuf

---

**EZY3735I**  
IUCV interrupt, fd=descriptor, path=path, type=type (string)

**Explanation**
This message is issued when socket tracing is requested and indicates an IUCV interrupt of the indicated type number (name) occurred for the file descriptor and path ID specified.
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
descarray_save_iucvbuf

---

**EZY3736S**  Had a rejected message completion, msg lost

**Explanation**
An IUCV interrupt denoting “rejected message” was received; the message is lost.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
descarray_save_iucvbuf

---

**EZY3737W**  Sever Pending Connection from addrspace for application

**Explanation**
An IUCV interrupt was received for a socket that does not exit. The connection to the indicated TCPIP address space is severed.

**System action**
TCPIP continues.

**Operator response**
None.
EZY3738W  Sever Pending Connection, no user data

Explanation
The pending connection of the IUCV path to the TCPIP address space has been severed. No user data is available for the connection.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

EZY3739W  Message Complete audit field: value.value.value

Explanation
An IUCV interrupt occurred with nonzero audit data. This signifies an error was detected after the IUCV operation was requested. The values in the message indicate more specific details about the type of error.

System action
Error return will be indicated by the socket function invoked.

Operator response
None.

System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the traces and the dump.
Module
CMIUCSOC

Procedure name
check_audit

EZY3740I  Rc=code on IUCV_function fd=descriptor, buf (ipbfadr2) is at address

Explanation
This message appears during a request to get the next IUCV buffer if socket tracing is in effect. The return code from the IUCV_NEXTBUFFER function, the IUCV function, the file descriptor, and the address of the buffer for the request are displayed.

This message is part of the socket trace used for problem analysis and can be generated by specifying trace socket in the TCPIP.DATA configuration file.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
doiuvc

EZY3741I  Rc=code on IUCV_function to id fd=descriptor, path=path, iprcode=iprcode, ipmsgid=hexid, iucvname=string

Explanation
This message displays information about an IUCV function including:
• Return code
• Function name
• User ID of the TCPIP address space
• File descriptor
• Path ID
• The isolated pacing response (IPR) code
• The message ID from the function
• IUCV name or descriptor name

This message is part of the socket trace used for problem analysis and can be generated by specifying trace socket in the TCPIP.DATA configuration file.
System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
doicv

**EZY3742I**  ciucv_data area (ipbfadr2) is at address

Explanation
The IPBFADR2 data area is at the location pointed to by the address in the message.
This message is part of the socket trace used for problem analysis and can be generated by specifying *trace socket* in the TCPIP.DATA configuration file.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
doicv

**EZY3743I**  Caught a REPLY after PURGE

Explanation
A reply external interrupt occurred after the IUCV_PURGE function. This message is generated only when socket tracing is in effect. IUCV ignores the reply.

System action
TCPIP continues.

Operator response
None.
System programmer response
None.

Module
CMIUCSOC

Procedure name
doiucv

**EZY3744E**  
*Rc=code on IUCV_function to id, path=path, iprcode=code, iucvname=name*

Explanation
This message displays information about an unsuccessful IUCV function including:

- Return code
- Function name
- User ID of the TCPIP address space
- Path ID
- The isolated pacing response (IPR) code
- IUCV name or descriptor name

This message appears when socket tracing is not in effect (when socket tracing is specified, other messages convey the same information as this one, therefore this message is not generated.)

System action
TCPIP continues.

Operator response
None.

System programmer response
If the error can be reproduced, set *trace socket* to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
CMIUCSOC

Procedure name
doiucv

**EZY3745E**  
*Should not be in sock_request_iucv for function,*

Explanation
The sock_request_iucv routine should not have been called for the function indicated in this message. IUCV returns a −1 return code and an error to indicate that an incorrect file number was detected.
System action
TCPIP continues.

Operator response
None.

System programmer response
This is an internal error. Contact the IBM Software Support Center to report the error.

Module
CMIUCSOC

Procedure name
sock_request_iucv

**EZY3746I** sock request_inet entry parms:

Explanation
This message precedes the list of parameters that are being used for the sock_request_inet routine, the routine that sends an INET socket request.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace socket` in the TCPIP.DATA configuration file.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
sock_request_inet

**EZY3747I** sock_request_inet returns EIBMIUCVERR

Explanation
The sock_request_inet routine encountered an error while trying to send an INET socket request. This message has additional text concatenated to it that gives more information about the circumstances where the error occurred.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace socket` in the TCPIP.DATA configuration file.
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
sock_request_inet

**EZY3748X  after check_and_establishiucv() err**

**Explanation**
This message is concatenated to message EZY3747I. The current IUCV path has been severed. The socket application attempted to establish a new path, but was not successful. The application sets the error number parameter to indicate an IUCV error, and the return code to indicate no path ID.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace socket` in the TCPIP.DATA configuration file.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
sock_request_inet

**EZY3750X  after doiucv() err**

**Explanation**
This message is concatenated to message EZY3747X. An error occurred when IUCV sent an INET socket request to TCPIP.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace socket` in the TCPIP.DATA configuration file.
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
sock_request_inet

---

**EZY3752X** due to bad irpt

**Explanation**
This message is concatenated to message EZY3747X. An error occurred when an IUCV interrupt was received. Information in the interrupt buffer does not match what was expected.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace` socket in the TCPIP.DATA configuration file.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
sock_request_inet

---

**EZY3754X** after check_audit()

**Explanation**
This message is concatenated to message EZY3747I. The check_audit routine checked the audit field of an IUCV interrupt and the field indicated an error occurred during the interrupt. The return code and errno value provide more information about the error. IUCV returns the return code and errno value in the parameter list.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace` socket in the TCPIP.DATA configuration file.
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
sock_request_inet

**EZY3755I sock_request_inet return parms:**

**Explanation**
These are the parameters that the sock_request_inet routine returns upon completion. If an error occurred during this routine, the return code and errno will appear in this parameter list.

This message is part of the socket trace used for problem analysis and can be generated by specifying `trace socket` in the TCPIP.DATA configuration file.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
sock_request_inet

**EZY3756I flush_dgram: about to call sock_request(), totlenbufferlength**

**Explanation**
The sock_request function is to be called, with a buffer of the length indicated in the message. This call will transmit all locally buffered datagrams to TCPIP. This message is generated only when a socket is operating in bulk mode. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.
**Explanation**
This is the number of datagrams being sent for a socket request. Message EZY3758I will be generated for each of these datagrams. This message is generated only when a socket is operating in bulk mode. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_send

**EZY3757I** sending iovcnt=\textit{number}

**Explanation**
The iovc structure fields (address and length of a datagram) is displayed, followed by the first 50 bytes of the datagram in hexadecimal format. This message is generated only when a socket is operating in bulk mode. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

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

None.

**Module**

CMIUCSOC

**Procedure name**

common_send

**EZY3774I  fd=descriptor in callers rmask**

**Explanation**

This message identifies a file descriptor set in the caller's read mask. The message will be issued once for each file descriptor that is set in the read mask. This message is generated only when socket tracing is in effect.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

CMIUCSOC

**Procedure name**

common_select

**EZY3775I  fd=descriptor unallocated**

**Explanation**

The indicated file descriptor is not allocated, but it was specified in a read, write, or exception mask supplied by an application as an argument to a select function. This message is generated only when socket tracing is in effect. The select function returns an error to indicate an incorrect file number.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

CMIUCSOC
**EZY3776I**  
fd=descriptor in callers wmask

**Explanation**
This message identifies the file descriptors set in the caller’s write mask. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

---

**EZY3777I**  
fd=descriptor in callers emask

**Explanation**
This message identifies the file descriptors set in the caller’s exception mask. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

---

**EZY3778I**  
Select would cause deadlock
Explanation
The arguments to select specified nothing (no file descriptor, no time-out, etc.) that could be or become ready, therefore an attempt to perform this select function would never return. Error code EDEADLK is returned to caller.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
common_select

EZY3779I in inetselect

Explanation
The status of the AF_INET descriptors in the inetselect mask is being checked (an interaction with TCPIP will occur). This message is generated only when socket tracing is in effect.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
common_select

EZY3780I in inetselect iucverror

Explanation
The socket IUCV path to the TCPIP address space no longer exists. (This might occur if TCPIP is unsuccessful, or is intentionally stopped, for example.) This message is generated only when socket tracing is in effect. Error code -1 is generated by the select function called by the application.
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

---

**EZY3781I**  
waitvalue of 0 secs to TCPIP

**Explanation**
A select function has already determined a selected AF_IUCV descriptor is ready. It is forcing a wait time of zero when contacting TCPIP to learn whether any selected AF_INET descriptor is also ready. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

---

**EZY3782I**  
in iucvselect, iucvnfds=numberfds

**Explanation**
This message indicates the number of ready AF_IUCV file descriptors found in the select mask. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

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

Module
CMIUCSOC

Procedure name
common_select

EZY3783I  user specified seconds s micro-seconds ms wait

Explanation
The user requested the specified time-out value in a call to a select function. This message is generated only when socket tracing is in effect.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
common_select

EZY3784I  issue stimer now

Explanation
A select function was called with a time-out argument, but no AF_INET descriptors in the selection mask and no other selected descriptors ready. Therefore, the select function will use STIMER to measure the wait time locally. This message is generated only when socket tracing is in effect.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC
**Procedure name**
common_select

**EZY3785I**  
do not issue stimer

**Explanation**
A select function was called with a time-out value of zero. No selected descriptor is ready, and no wait will be performed. The select function sets the user file descriptor masks to zero (no descriptor is ready) and returns a zero return code. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

**EZY3786I**  
wait forever, null ptr to timeval

**Explanation**
A select function has been called with NULL as the time-out argument. No selected descriptor is currently ready, therefore the socket application will wait until a descriptor becomes ready (or the specified ECB is posted, if selectex was called). This message is generated only when socket tracing is in effect.

**System action**
The application waits until the wanted action occurs.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

**EZY3787I**  
wait ecblist=list, ecbcound=count

EZY3xxx messages 373
**Explanation**
Displays the address of the ECB list, and the number of ECBs in this list, that will be used by a select function to wait for an event to occur. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

**EZY3788I** iucvposted=iucv, waitposted=wait, callposted=call

**Explanation**
Describes the result of the WAIT operation described by message EZY3787I, specifically indicating the posted status of the ECB used for IUCV interrupts (iucvposted), the ECB used for timer interrupts (waitposted), and the ECB supplied by the calling application (callposted). This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

**EZY3789I** in iucvposted

**Explanation**
A select function is processing an IUCV interrupt. If the interrupt indicates a message complete from TCPIP, IUCV merges the AF_IUCV file descriptor sets with the AF_INET file descriptor sets returned by TCPIP. If there are AF_IUCV descriptors in the caller’s masks, any IUCV interrupt selected by the caller’s mask completes the request and the caller’s mask is set accordingly. If an AF_IUCV interrupt is received for a descriptor not in the
caller's mask, the select function continues waiting. This message is generated only when socket tracing is in effect.

**System action**
The application's action depends on the specific case. See the explanation for more information. TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

---

**EZY3790I** iucvcomp is now TRUE

**Explanation**
A select function found at least one AF_IUCV descriptor in the caller's masks, and an IUCV interrupt occurred on an AF_IUCV path. Assignment of TRUE to the variable iucvcomp causes later code to check whether this IUCV interrupt satisfies the select condition. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

---

**EZY3791I** in gotmsgcomp

**Explanation**
This indicates that select received a Message Complete interrupt from TCPIP (because one of the AF_INET descriptors identified by the select function's caller is ready) and is about to proceed accordingly. Select adds any AF_IUCV descriptors that are ready to the AF_INET ready descriptors in the mask, and continues. This message is generated only when socket tracing is in effect.
**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

**EZY3792I  nfds=value, return=value**

**Explanation**
Displays the number of AF_IUCV descriptors that are ready (nfds) and the total number of ready descriptors (return), which is the value returned by select. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

**EZY3793I  must wait for TCPIP**

**Explanation**
This indicates that a select function must wait for a reply to its message to TCPIP before it can complete. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.
**EZY3794I** in iucvcom && iucvselect

**Explanation**
A select function is ending because an AF_IUCV descriptor has become ready before any AF_INET descriptor or other action that would end the select (for example, time-out or ECB posting). This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

**EZY3795I**  in wait posted

**Explanation**
A select function is ending because the time-out value specified by the caller has expired. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC
**Procedure name**
common_select

**EZY3796I**  in wait posted iucvselect

**Explanation**
A select function’s time-out period has expired (this message follows message EZY3795I) and the selection mask includes one or more AF_IUCV descriptors. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

**EZY3797I**  callers ECB posted

**Explanation**
A select function is ending because the ECB identified by the caller has been posted, rather than any of the specified descriptors becoming ready. Select sets the caller’s masks to zero and returns a zero return code.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
common_select

**EZY3818I**  Your dataset has enabled debug.
**Explanation**
Debugging has been requested by an option in your TCPIP.DATA data set. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
ReadProf

**EZY3819I**  Your dataset has enabled TESTSTOR.

**Explanation**
The TESTSTOR function has been requested by an option in your TCPIP.DATA data set. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
CMIUCSOC

**Procedure name**
ReadProf

**EZY3820I**  Your dataset has disabled TESTSTOR.

**Explanation**
The TESTSTOR function has been turned off by an option in your TCPIP.DATA data set. This message is generated only when socket tracing is in effect.

**System action**
TCPIP continues.
EZY3821I  TESTSTOR is enabled.

Explanation
The TESTSTOR function is active and will check the validity of storage addresses provided as arguments in calls to socket functions. This message is generated only when socket tracing is in effect.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
CMIUCSOC

Procedure name
ReadProf

EZY3822I  TESTSTOR is not enabled.

Explanation
The TESTSTOR function is not active. This message is generated only when socket tracing is in effect.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.
Module
CMIUCSOC

Procedure name
ReadProf

EZY3950X No error

Explanation
This message indicates normal completion, no Inter-User Communication Vehicle (IUCV) error is found. The socket call is successful.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
TCPERROR

Procedure name
tcperror

EZY3951X Invalid Path-id

Explanation
The path ID in an IUCV operation is incorrect. The socket call ends.

System action
TCPIP continues.

Operator response
None.

System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR
**EZY3952X  Path is quiescent**

**Explanation**
The IUCV path used in a send operation has been quiesced by the receiver. No messages can be sent on this path until the receiver has performed a resume operation.

**System action**
The socket call continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set `trace socket` to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error.

After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**EZY3953X  Msg limit Exceeded**

**Explanation**
The maximum number of IUCV messages has already been queued for the receiver on the path over which a new message send request has been made. The IUCV message is discarded. An error return is generated by the socket function the application called.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set `trace socket` to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error.

After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR
**Procedure name**
tcerror

**EZY3954X  Priority msg not allowed on this path**

**Explanation**
The IUCV CONNECT parameter for the connection listed priority = NO, so no priority messages are allowed on this path. The message is discarded.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set `trace socket` to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY3955X  Reply too big**

**Explanation**
An IUCV SEND operation specified a reply was required, but the length of the reply sent by the message recipient exceeded the size of the answer buffer provided by the message sender for this reply.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set `trace socket` to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR
**EZY3956X**  Fetch Protection Exception on answer buffer

**Explanation**
The address specified for the reply message by an IUCV REPLY operation points to an address in storage that is fetch protected.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcpreor

---

**EZY3957X**  Addressing Exception on answer buffer

**Explanation**
The pointer to the reply buffer specified in an IUCV operation is incorrect or null.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR
**Procedure name**
tcerror

**EZY3958X  Conflicting msg class\path\msgid**

**Explanation**
This message indicates one of three conditions:

- an unrecognized socket call constant was found in the high-order halfword of the target message class (TRGCLS) keyword.
- conflicting path ID
- conflicting message ID

IUCV operation is unsuccessful.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error.

After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY3959X  Msg was purged**

**Explanation**
An IUCV REPLY was performed for a message that has been purged by the sender. The reply message is discarded.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error.
After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR

Procedure name
tcperro

EZY3960X Negative msg length

Explanation
The buffer length (BUFLEN) or data length (PRMMSG) is negative and is therefore an incorrect parameter for an IUCV operation. The IUCV call ends.

System action
TCPIP continues.

Operator response
None.

System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR

Procedure name
tcperro

EZY3961X Target userid not logged on

Explanation
The target communicator specified for an IUCV CONNECT function does not exist. The IUCV call ends.

System action
TCPIP continues.

Operator response
None.

System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error.
After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcperro

**EZY3962X**  Target userid not enabled for IUCV

**Explanation**
The target communicator specified for an IUCV CONNECT function has not enabled IUCV operations. The IUCV call ends.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcperro

**EZY3963X**  Exceeded number of paths (this side)

**Explanation**
You have exceeded the maximum of 255 simultaneous connections allowed under IUCV. The connection is not established.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might
suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZY3964X  Exceeded number of paths (other side)**

**Explanation**
The target communicator has exceeded the maximum of 255 simultaneous connections allowed under IUCV. The IUCV call ends.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZY3965X  Not authorized**

**Explanation**
You are not authorized to establish an IUCV connection with the remote user’s address space. The IUCV call ends.

**System action**
TCPIP continues.

**Operator response**
None.
System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR

Procedure name
tcerror

EZY3967X Invalid function code

Explanation
Unknown function specified for IUCV operation. No operation is performed.

System action
TCPIP continues.

Operator response
None.

System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR

Procedure name
tcerror

EZY3968X Msg limit exceeds 255

Explanation
The message limit specified for an IUCV CONNECT operation is too large. The IUCV CONECT operation is ignored.

System action
TCPIP continues.

Operator response
None.
**System programmer response**

If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**

TCPERROR

**Procedure name**

tcpper

**EZY3969X** Buffer already declared

**Explanation**

An IUCV buffer had already been declared, the first step in initializing for IUCV operations, when another IUCV DECLARE BUFFER request was made.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**

TCPERROR

**Procedure name**

tcpper

**EZY3970X** Path was severed by other side

**Explanation**

The IUCV partner has severed the path for which an IUCV ACCEPT function is requested. The ACCEPT function is unsuccessful.

**System action**

TCPIP continues.

**Operator response**

None.
System programmer response

If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR

Procedure name
tcperror

EZY3971X Cannot accept data in parmlist

Explanation
An IUCV SEND operation specifying data to be delivered in the parameter list could not be performed because the CONNECT operation that established the IUCV path to be used for the SEND operation did not specify that data could be delivered in the parameter list. The IUCV operation is unsuccessful.

System action
TCPIP continues.

Operator response
None.

System programmer response

If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR

Procedure name
tcperror

EZY3972X SEND buffer list is invalid

Explanation
The send buffer list data is incorrect or null. The IUCV operation is unsuccessful.

System action
TCPIP continues.

Operator response
None.
System programmer response
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCERROR

**Procedure name**
tcerror

**EZY3973X  ** Negative length in answer list

**Explanation**
The length specified in the IUCV socket call for the buffer list (BUFLEN) is negative. The socket call ends.

**System action**
TCPIP continues.

**Operator response**
None.

System programmer response
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCERROR

**Procedure name**
tcerror

**EZY3974X  ** Total list length is invalid

**Explanation**
A buffer list was specified for an IUCV SEND or REPLY operation, and the total message length specified does not match the sum of the individual lengths of the message pieces in the buffer list. The IUCV call ends.

**System action**
TCPIP continues.

**Operator response**
None.
System programmer response
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZ3975X**  PRMMSG and BUF/ANSLIST conflict

**Explanation**
The PRMMSG option is not allowed when you specify ANSLIST = YES. The message is not sent.

**System action**
TCPIP continues.

**Operator response**
None.

System programmer response
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZ3976X**  Buffer list not on double word boundary

**Explanation**
The buffer list for an IUCV call must begin on an appropriate boundary in storage. The IUCV call is unsuccessful.

**System action**
TCPIP continues.

**Operator response**
None.
System programmer response

If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**

TCPERROR

**Procedure name**

tcerror

**EZY3977X**  
**Answer list not on double word boundary**

**Explanation**

The answer list for an IUCV call must begin on an appropriate boundary in storage. Like the BUFLIST field (see message EZY3976X), the ANSLIST must begin on a double-word boundary. The IUCV call is unsuccessful.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**

TCPERROR

**Procedure name**

tcerror

**EZY3978X**  
**Must create control buffer**

**Explanation**

An attempt was made to perform an IUCV operation, such as CONNECT, on a control path but no DECLARE BUFFER has been issued for control paths. All IUCV operations default to application path.

**System action**

TCPIP continues.

**Operator response**

None.
System programmer response

If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR

Procedure name
tcperror

EZY3985E Bad socket-call constant (EIBMBADCALL)

Explanation
An incorrect socket-call constant was found in the IUCV header, such as a code that refers to a non-existent socket function. The socket call is unsuccessful.

System action
TCPIP continues.

Operator response
None.

System programmer response
Correct the arguments in the socket call. For details, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror

EZY3986E Bad parm (EIBMBADPARM)

Explanation
One of the parameters specified in the socket call is incorrect. The socket call is unsuccessful.

System action
TCPIP continues.
**Operator response**

None.

**System programmer response**

Correct the offending parameter descriptor. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**

TCPERROR

**Procedure name**

tcperror

**EZY3987E**  Socket out of range (EIBMSOCKOUTOFRANGE)

**Explanation**

The socket descriptor is outside the range accepted for the IBM implementation of sockets. The socket call is unsuccessful.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

Change the socket descriptor for the IBM socket to an acceptable value. For details, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**

TCPERROR

**Procedure name**

tcperror

**EZY3988E**  Socket in use (EIBMSOCKINUSE)

**Explanation**

The socket number assigned by the client application for accept, socket, or takesocket call is already in use. The socket call is unsuccessful.

**System action**

TCPIP continues.

**Operator response**

None.
System programmer response
Create a new accept, socket, or takesocket call for the application. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR

Procedure name
tcperor

EZY3989E  IUCV error (EIBMUCVERR)

Explanation
The request failed due to an IUCV error. This error is generated by the client stub code. The socket call is unsuccessful.

System action
TCPIP continues.

Operator response
None.

System programmer response
Use the IUCV error message that appears directly after this message to aid in determining the nature of the error. For more information about IUCV sockets, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperor

EZY3993E  Conflict with other call on socket (EIBMCONFLICT)

Explanation
A previous function on the same socket has not completed and the present call is not of a type that can be processed or queued until the previous operation completes. The present call is unsuccessful.

System action
TCPIP continues.

Operator response
None.
System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR

Procedure name
tcperro

EZY3994E  Call canceled (EIBMCANCELLED)

Explanation
A close() call has been issued by either the server or the client.

System action
If the socket is closed when input data is queued, the TCP connection is reset rather than being cleanly closed.

Operator response
None.

System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message, the traces, and the dump.

Module
TCPERROR

Procedure name
tcperro

EZY3996E  A TCP/IP name that is not valid was detected (EIBMADTCPNAME)

Explanation
A TCP/IP name that was not valid was detected. The socket call is unsuccessful.

System action
TCP/IP continues.

Operator response
None.
System programmer response
The system derives the TCP/IP value from the configuration file, as described in the z/OS Communications Server: IP Configuration Reference. Correct the TCPIPJOBNAME value defined in the TCPIP.DATA file that is allocated to the job.

Module
TCPERROR

Procedure name
tcperror

EZY3997E TSRB request code is invalid (EIBMBADREQUESTCODE)

Explanation
A request code that was not valid was detected.

System action
TCP/IP continues.

Operator response
None.

System programmer response
Correct the arguments in the socket call. For details, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. After the error occurs, obtain a dump of the VMCF address space. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror
Chapter 6. EZY4xxx messages

**EZY4000E**  A connection token that is not valid was detected. (EIBMBADCONNECTIONMATCH)

**Explanation**
A connection token that is not valid was detected. There is no such connection.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZY4001E**  TCPIP Abend (EIBMTCPABEND)

**Explanation**
An abend occurred in TCP processing of this request.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR
**EZY4008E**  TCPIP C Socket Abend (EIBMTERMERROR)

**Explanation**  
TCP/IP C Socket encountered a terminating error.

**System action**  
TCP/IP continues.

**Operator response**  
None.

**System programmer response**  
If the error can be reproduced, set `trace socket` to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option `SOCKET` active. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**  
TCPPERROR

**EZY4011E**  The delete requestor did not create the connection. (EIBMINVDELETE)

**Explanation**  
The request was called from an invalid task.

**System action**  
TCP/IP continues.

**Operator response**  
None.

**System programmer response**  
If the error can be reproduced, set `trace socket` to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option `SOCKET` active. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**  
TCPPERROR
**Procedure name**

tcperror

**EZY4012E** The connection token validation failed. (EIBMINVSOCKET)

**Explanation**
Connection token validation failed. The socket does not exist.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

**Procedure name**

tcperror

**EZY4013E** Connection Terminated by TCPIP (EIBMINVTCPCONNECTION)

**Explanation**
A socket connection was terminated by TCP/IP.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR
Procedure name
tcperror

**EZY4021E** TCPIP not installed or not active (EIBMNOACTIVETCP)

**Explanation**
TCPIP is not installed or not active.

**System action**
TCPIP not available.

**Operator response**
Issue a D TCPIP command to determine whether TCPIP is installed. If it is installed, restart TCPIP and reissue the socket call.

**System programmer response**
None.

**Module**
TCPERROR

Procedure name
tcperror

**EZY4022E** The requested control block contained data that is not valid. (EIBMINVTSRBUSERDATA)

**Explanation**
The requested control block contained data that is not valid.

**System action**
TCP/IP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set **trace socket** to active in the TCPIP.DATA configuration file and rerun the application. The traces provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message, the traces, and the dump.

**Module**
TCPERROR

Procedure name
tcperror

**EZY4026E** Client received bad post code (EIBMBADPOSTCODE)
Explanation
This message is received by an outstanding (blocked) socket when TCPIP is stopped. It indicates that an application was given a bad post code.

System action
TCPIP ends.

Operator response
None.

System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces will provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror

EZY4050E No error

Explanation
This message indicates normal completion of a socket call.

System action
TCPIP continues.

Operator response
None.

System programmer response
If the application is correct in detecting that some error occurred, another program has reset the value of errno before the call to tcperror generated this message. Otherwise, the application logic has incorrectly decided an error occurred, and is calling tcperror to display a message when no reason exists to do so.

Module
TCPERROR

Procedure name
tcperror

EZY4051E Not owner (EPERM)

Explanation
Permission is denied. The caller does not have correct ownership to perform the requested operation.
System action
The call fails. TCPIP continues.

Operator response
None.

System programmer response
Check the permissions for the data set or directory.

Module
TCPERROR

Procedure name
tcperro

EZY4052E  No such file or directory (ENOENT)

Explanation
A referenced data set or directory does not exist.

System action
TCPIP continues.

Operator response
None.

System programmer response
Verify the data set or directory name, and correct the reference.

Module
TCPERROR

Procedure name
tcperro

EZY4053E  No such process (ESRCH)

Explanation
The referenced process or task was not found.

System action
TCPIP continues.

Operator response
None.
**System programmer response**
Verify the process name, and correct the reference.

**Module**
TCPERROR

**Procedure name**
tcpenror

**EZY4054E  Interrupted system call (EINTR)**

**Explanation**
A function was unable to complete, usually because of conditions having nothing to do with the request itself. This code does not diagnose an error in a request, but reports that the system handling the request found it more convenient to abandon this request than to continue it after some other activity was performed.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
Repeat the original request.

**Module**
TCPERROR

**Procedure name**
tcpenror

**EZY4055E  I/O error (EIO)**

**Explanation**
Some sort of I/O error occurred while the operation was in progress. This could be a disk error, for example.

**System action**
TCPIP continues.

**Operator response**
None.

**System programmer response**
Make sure that sufficient storage exists for newly-created data sets. Use the appropriate I/O support functions under MVS to isolate the source of the error.
**Procedure name**
tcperro

**EZY4056E**  No such device or address (ENXIO)

**Explanation**
The process name or driver name for the socket call is not found.

**System action**
The socket call fails. TCPIP continues.

**Operator response**
None.

**System programmer response**
Verify the local and remote addresses and process names for the socket call. For more information about socket calls, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**
TCPERROR

**Procedure name**
tcperro

**EZY4057E**  Arg list too long (E2BIG)

**Explanation**
The argument list for the call is too long.

**System action**
The socket call fails. TCPIP continues.

**Operator response**
None.

**System programmer response**
Correct the arguments to the call. Reinitialize the application.

If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**
TCPERROR

**Procedure name**
tcperro
EZY4058E   Exec format error (ENOEXEC)

Explanation
The sequence of calls in the client or server program is incorrect. The socket call cannot be executed as written.

System action
The socket call fails. TCPIP continues.

Operator response
None.

System programmer response
Check the sequence of calls within the application program. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.
If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror

EZY4059E   Bad file number (EBADF)

Explanation
An invalid argument to the socket call was specified. The nature of the error depends on the particular call that triggers it.

System action
The socket call fails. TCPIP continues.

Operator response
None.

System programmer response
Correct the parameters of the appropriate socket call. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for information about System Error Return Codes.
If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror

EZY4060E   No children (ECHILD)
Explanation
No daughter processes exist. Generated by some X-Window System functions.

System action
TCPIP continues.

Operator response
None.

System programmer response

Module
TCPERROR

Procedure name
tcerror

EZY4061E  No more processes (EAGAIN)

Explanation
There are no more processes queued to this connection.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
TCPERROR

Procedure name
tcerror

EZY4062E  Not enough memory (ENOMEM)

Explanation
A memory allocation (malloc) call for an operation failed.

System action
The socket call fails. TCPIP continues.

Operator response
None.
System programmer response
Correct the parameters of the appropriate socket call. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for information about System Error Return Codes.
If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcerror

EZY4063E  Permission denied (EACCES)

Explanation
The caller does not have the correct permissions to perform the requested operation.

System action
The request fails. TCPIP continues.

Operator response
None.

System programmer response
Correct the parameters of the appropriate socket call. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for information about System Error Return Codes.
If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcerror

EZY4064E  Bad address (EFAULT)

Explanation
An incorrect storage address or length was specified in the socket call. The following list displays some of the possible function calls that might return this error value:

<table>
<thead>
<tr>
<th>Call</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>accept</td>
<td>Using ADDR and ADDRLEN would result in an attempt to copy the address into a portion of the caller's address space into which information cannot be written.</td>
</tr>
<tr>
<td>bind</td>
<td>Using NAME and NAMELEN would result in an attempt to copy the address into a nonwritable portion of the caller's address space.</td>
</tr>
<tr>
<td>connect</td>
<td>Using NAME and NAMELEN would result in an attempt to copy the address into a portion of the caller's address space into which data cannot be written.</td>
</tr>
</tbody>
</table>
**Call**

send

takesocket

**Explanation**

Using the MSG and LEN parameters would result in an attempt to access storage outside the caller's address space.

Using the CLIENTID parameter as specified would result in an attempt to access storage outside the caller's address space.

**System action**

The socket call fails. TCPIP continues.

**Operator response**

None.

**System programmer response**

Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this to your application and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**

TCPERROR

**Procedure name**

tcperror

**EZY4065E Block device required (ENOTBLK)**

**Explanation**

An inappropriate kind of I/O device was specified for an operation.

**System action**

TCPIP continues.

**Operator response**

None.

**System programmer response**

**Module**

TCPERROR

**Procedure name**

tcperror

**EZY4066E Device busy (EBUSY)**
**Explanation**
This message indicates that listen() has already been called for the socket.

**System action**
The socket call ends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Correct or delete the givesocket() call. Try the task again.
If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**
TCPERROR

**Procedure name**
tcpererror

---

**EZY4067E**  
**File exists (EEXIST)**

**Explanation**
The specified data set already exists. Your socket call would overwrite it.

**System action**
The socket call ends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Change the write() call.
If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**
TCPERROR

**Procedure name**
tcpererror

---

**EZY4068E**  
**Cross-device link (EXDEV)**

**Explanation**
A socket call specified a cross-device link.
System action
The socket call ends. TCPIP continues.

Operator response
None.

System programmer response
Respecify the parameters of the appropriate socket call to prevent the cross-device link.
If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror

EZY4069E  No such device (ENODEV)

Explanation
The device pointed to in the givesocket() does not exist.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Correct the client ID parameter in the givsocket() call.
If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror

EZY4070E  Not a directory (ENOTDIR)

Explanation
The caller specified a non-directory in a directory operation.

System action
The socket call abends. TCPIP continues.
**Operator response**
None.

**System programmer response**
Correct the pointer in the socket call parameter.

If you are still unable to determine the cause of the problem, create the problem again with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**
TCPERROR

**Procedure name**
tcperor

**EZY4071E**  Is a directory (EISDIR)

**Explanation**
The caller specified a directory in a non-directory operation.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Correct the pointer in the socket call parameter.

**Module**
TCPERROR

**Procedure name**
tcperor

**EZY4072E**  Invalid argument (EINVAL)

**Explanation**
The argument specified to the socket call was not valid. The nature of the error depends on the particular call that triggers it. The following list displays some of the possible function calls that might return this error value:

- **accept**
  Listen was not called for this socket.

- **bind**
  The socket is already bound to an address.

- **connect**
  The specified name length is incorrect.

- **fcntl**
  Incorrect flags were specified.
givesocket
An incorrect client ID was entered.

ioctl
The request is incorrect or not supported.

select
One of the fields in the time-out structure is incorrect.

sendto
The target address length is incorrect for the specified address family.

shutdown
The shutdown condition is not 0, 1, or 2.

takesocket
The specified client ID is incorrect.

System action
The socket call ends. TCPIP continues.

Operator response
None.

System programmer response
Use the information provided in the above table as a guide for the possible conditions under which this error value can occur. Apply this to your application and, correct the parameters of the appropriate socket call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR

Procedure name
tcperror

EZY4073E File table overflow (ENFILE)

Explanation
The data set containing the connection table for the TCPIP address space exceeded its maximum size.

System action
The socket call ends. TCPIP continues.

Operator response
None.

System programmer response
Use the maxdesc() call to increase the maximum number of sockets from the default value of 47. Then use the getdtablesize() call to verify the new maximum. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR
**EZY4074E**  Too many open files (EMFILE)

**Explanation**
The socket descriptor table is already full. The following list displays some of the possible function calls that might return this error value:

- **maxdesc**
  Indicates that *totdesc is greater than 2000.

- **takesocket**
  The socket descriptor table is already full.

**System action**
The call stops. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

---

**EZY4075E**  Inappropriate ioctl for device (ENOTTY)

**Explanation**
An incorrect device call was specified in the cmd parameter of the ioctl() call.

**System action**
The socket call closes. TCPIP continues.

**Operator response**
None.

**System programmer response**
Change or delete the ioctl() command for the device. For more information about the ioctl() call, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR
EZY4076E  Text file busy (ETXTBSY)

Explanation
The text data set pointed to by the call is currently being written to by another process.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Try the socket call again.

Module
TCPERROR

Procedure name
tcperror

EZY4077E  File too large (EFBIG)

Explanation
The data set is too large. The operation caused the data set to grow beyond the server’s limit.

System action
The socket call ends. TCPIP continues.

Operator response
None.

System programmer response
Change the protocol or socket type specified in the socket() call. For more information about the socket() call, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR

Procedure name
tcperror

EZY4078E  No space left on device (ENOSPC)

Explanation
Insufficient storage exists on the target device to carry out the operation. The operation caused the server’s file system to reach its limit.
System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Correct the len parameter of the send() call. For more information about the send() call, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR

Procedure name
tcperror

EZY4079E  Illegal seek (ESPIPE)

Explanation
The stream pipe specified in the socket call is not correct.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Correct the read() call parameters. For more information about the read() call, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR

Procedure name
tcperror

EZY4080E  Read-only file system (EROFS)

Explanation
Your application cannot write to the data set specified in the socket call. The data set resides on a read-only file system.

System action
The socket call abends. TCPIP continues.

Operator response
None.
System programmer response
None.

Module
TCPERROR

Procedure name
tcperror

EZY4081E Too many links (EMLINK)

Explanation
The host connection queue is already full.

System action
The call is bypassed. TCPIP continues.

Operator response
None.

System programmer response
Try the call again.

Module
TCPERROR

Procedure name
tcperror

EZY4082E Broken pipe (EPIPE)

Explanation
One of the stream pipes used to redirect I/O in the socket call has failed.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Perform a SOCKET or IUCV trace to determine where the failure occurred. For more information, see z/OS Communications Server: IP Diagnosis Guide.
Procedure name
tcerror

**EZY4083E**   Argument too large (EDOM)

**Explanation**
The parameter for the socket call was too large to be a valid argument to the call.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Correct the socket call parameter. For more information, see *z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference*.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY4084E**   Result too large (ERANGE)

**Explanation**
The value returned by the socket call is out of the range of anticipated values.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Correct the socket call parameters as necessary and try the call again.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY4085E**   Operation would block (EWOULDBLOCK)

**Explanation**
The meaning and severity of this message depend on the call that generates it. The following list displays some of the possible function calls that might return this error value:
accept
The socket is in nonblocking mode and connections are not queued. This is not an error condition.

read recvfrom
The socket is in nonblocking mode and read data is not available. This is not an error condition.

send sendto write
The socket is in nonblocking mode and buffers are not available.

System action
The socket call is bypassed. TCPIP continues.

Operator response
None.

System programmer response
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

EZY4086E Operation now in progress (EINPROGRESS)

Explanation
The socket is marked nonblocking and the connection cannot be completed immediately. The following list displays some of the possible function calls that might return this error value:

connect
The socket descriptor s is marked nonblocking, and the connection cannot be completed immediately. The EINPROGRESS value does not indicate an error condition.

System action
The socket call is bypassed. TCPIP continues.

Operator response
None.

System programmer response
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this to your application and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.
**EZY4087E Operation already in progress (EALREADY)**

**Explanation**
The operation specified in the socket call is already in progress. The following list displays some of the possible function calls that might return this error value:

- **connect**
  The socket descriptor `s` is marked nonblocking, and a previous connection attempt has not completed.

- **maxdesc**
  Your program called maxdesc() after creating a socket, or after a previous call to maxdesc().

**System action**
The socket call is closed. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

---

**EZY4088E Socket operation on non-socket (ENOTSOCK)**

**Explanation**
A socket operation was requested on a non-socket. The socket was incorrectly defined.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Correct the socket call parameters.

**Module**
TCPERROR
EZY4089E  Destination address required (EDESTADDRREQ)

Explanation
A destination address is needed for the socket call to complete.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Add the destination address to the socket call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR

Procedure name
tcerror

EZY4090E  Message too long (EMSGSIZE)

Explanation
The message length specified in the sendto call is too large. The maximum is 32,767. The following list displays some of the possible function calls that might return this error value:

sendmsg
The message was too big to be sent as a single datagram. The default is large-envelope-size.

sendto
The message was too big to be sent as a single datagram. The default is large-envelope-size.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR
Procedure name
tcerror

EZY4091E  Protocol wrong type for socket (EPROTOTYPE)

Explanation
The specified protocol is incorrect for this socket type.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Change the protocol parameter for the socket call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR

Procedure name
tcerror

EZY4092E  Option not supported by protocol (ENOPROTOOPT)

Explanation
The socket option specified is incorrect for the protocol type or the level is not SOL_SOCKET. The following list displays some of the possible function calls that might return this error value:

getsockopt
The OPTNAME parameter is unrecognized, or the LEVEL parameter is not SOL_SOCKET.

setsockopt
The OPTNAME parameter is unrecognized, or the LEVEL parameter is not SOL_SOCKET.

getibmsockopt
The OPTNAME parameter is unrecognized, or the LEVEL parameter is not SOL_SOCKET.

setibmsockopt
The OPTNAME parameter is unrecognized, or the LEVEL parameter is not SOL_SOCKET.

System action
The socket call ends. TCPIP continues.

Operator response
None.

System programmer response
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.
EZY4093E  Protocol not supported (EPROTONOSUPPORT)

Explanation
The protocol specified in the socket call is not supported by the address domain or the socket type. The following list displays some of the possible function calls that might return this error value:

socket
The PROTOCOL is not supported in this domain, or this PROTOCOL is not supported for this socket type.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

EZY4094E  Socket type not supported (ESOCKTNOSUPPORT)

Explanation
The socket type specified in the socket call is not supported.

System action
The socket call ends. TCPIP continues.

Operator response
None.

System programmer response
Change the type parameter in the socket() call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.
**EZY4095E** Operation not supported on socket (EOPNOTSUPP)

**Explanation**
The selected socket is not a stream socket. The following list displays some of the possible function calls that might return this error value:

- **accept**
  - The `s` parameter is not of type `SOCK_STREAM`.

- **givesocket**
  - The socket type is not `SOCK_STREAM`.

- **listen**
  - The `s` parameter is not a socket descriptor that supports the `listen()` call.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

---

**EZY4096E** Protocol family not supported (EPFNOSUPPORT)

**Explanation**
The protocol specified in the socket call is not supported by the address family specified. The following list displays some of the possible function calls that might return this error value:

- **getclientid**
  - The domain is not AF_INET.

- **takesocket**
  - The domain field of the CLIENTID parameter is not AF_INET.

**System action**
The socket call ends. TCPIP continues.
**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

<table>
<thead>
<tr>
<th>Module</th>
<th>TCPERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Procedure name</td>
<td>tcperror</td>
</tr>
</tbody>
</table>

**EZY4097E  Address family not supported by protocol family (EAFNOSUPPORT)**

**Explanation**
The address family specified in the socket call is not supported by the protocol specified. It must be either AF_IUCV or AF_INET. The protocol field must be set to 0 if the domain parameter is set to AF_IUCV. The following table displays some of the possible function calls that might return this error value:

- **bind**
  The address family is not supported (it is not AF_IUCV or AF_INET).

- **connect**
  The address family is not supported.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

<table>
<thead>
<tr>
<th>Module</th>
<th>TCPERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Procedure name</td>
<td>tcperror</td>
</tr>
</tbody>
</table>

**EZY4098E  Address already in use (EADDRINUSE)**

**Explanation**
The bind() call cannot bind the address specified to the socket with descriptor s because the address is already in use. The following table displays some of the possible function calls that might return this error value:
**bind**
The address is already in use. See the SO_REUSEADDR option described under getsockopt() and the SO_REUSEADDR option described under setsockopt() in the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for more information. This error will also occur if the port specified on the bind has been configured as RESERVED in a TCP/IP profile port reservation statement.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY4099E Can't assign requested address (EADDRNOTAVAIL)**

**Explanation**
Either the address specified in the socket call is incorrect for the bind() call or the calling host cannot reach the address specified in the connect() call. The following list displays some of the possible function calls that might return this error value:

**bind**
The address specified is not valid on this host. For example, the internet address does not specify a valid network interface.

**connect**
The calling host cannot reach the specified destination.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters for the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.
Module
TCPERROR

Procedure name
tcperror

**EZY4100E**  Network is down (ENETDOWN)

**Explanation**
The host network is down.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Try the application again when the host network restarts.

Module
TCPERROR

Procedure name
tcperror

**EZY4101E**  Network is unreachable (ENETUNREACH)

**Explanation**
The network of the address specified in the name parameter of the connect() call cannot be reached from this host. The following list displays some of the possible function calls that might return this error value:

- **bind**
  The network cannot be reached from this host.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR
**Procedure name**
tcerror

**EZY4102E**  Network dropped connection on reset (ENETRESET)

**Explanation**
The connection was dropped when the socket was reset.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Try the socket call again.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY4103E**  Software caused connection abort (ECONNABORTED)

**Explanation**
The application program caused the connection to abend.

**System action**
The socket call ends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use sock_debug() to determine why the connection abended. For more information about tracing and debugging socket calls, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY4104E**  Connection reset by peer (ECONNRESET)
**Explanation**  
If a stream socket is closed by the remote process while there is input data queued, the TCP connection is reset rather than being cleanly closed.

**System action**  
The connection is reset.

**Operator response**  
None.

**System programmer response**  
The socket call is suspended.

**Module**  
TCPERROR

**Procedure name**  
tcperror

**EZY4105E  No buffer space available (ENOBUFS)**

**Explanation**  
The specific meaning of this message depends on the call that triggers it. The following list displays some of the possible function calls that might return this error value:

- **all**  
  No buffer space is available.

- **accept**  
  Not enough buffer space is available to create the new socket.

- **send sendto write**  
  Not enough buffer space is available to send the new message.

- **takesocket**  
  There is a socket control block (SCB) or socket interface control block (SKCB) shortage in the TCPIP address space.

**System action**  
The socket call ends. TCPIP continues.

**Operator response**  
None.

**System programmer response**  
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the [z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference](https://www.ibm.com).
**Procedure name**
tcperror

**EZY4106E  Socket is already connected (EISCONN)**

**Explanation**
The socket specified by the socket descriptor, s, in the connect() call is already in use. The following list displays some of the possible function calls that might return this error value:

<table>
<thead>
<tr>
<th>Call</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>connect</td>
<td>The socket descriptor s is already connected.</td>
</tr>
</tbody>
</table>

**System action**
The socket call continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZY4107E  Socket is not connected (ENOTCONN)**

**Explanation**
The socket specified by the descriptor, d, in the givesocket() call is not connected. The following list displays some of the possible function calls that might return this error value:

- getpeername
  - The socket is not in the connected state.

- givesocket
  - The socket is not connected.

**System action**
The call is unsuccessful.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.
information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

**Procedure name**
tcperror

---

**EZY4108E** Can’t send after socket shutdown (ESHUTDOWN)

**Explanation**
A send() call cannot be carried out after a shutdown() call.

**System action**
The socket call fails. TCPIP continues.

**Operator response**
None.

**System programmer response**
Rewrite the socket call so that the shutdown() follows the send(). The only call that can follow the shutdown() is the close() for the socket.

---

**Module**
TCPERROR

**Procedure name**
tcperror

---

**EZY4109E** Too many references: can’t splice (ETOOMANYREFS)

**Explanation**
A socket call had too many references. The splice could not be completed.

**System action**
The socket call fails. TCPIP continues.

**Operator response**
None.

**System programmer response**
Rewrite the socket call to reduce the number of references.
**EZY4110E  Connection timed out (ETIMEDOUT)**

**Explanation**
The connection time to live expired before the socket call was completed. The following list displays some of the possible function calls that might return this error value:

- **connect**
  The connection establishment timed out before a connection was made.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

---

**EZY4111E  Connection refused (ECONNREFUSED)**

**Explanation**
The connection request was rejected by the remote server. The following table displays some of the possible function calls that might return this error value:

- **connect**
  The connection request was rejected by the destination host.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the table above as a guide to the conditions under which this error can occur. Apply this information to your application, and correct the parameters of the appropriate socket call. For more information, see the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.
Module
TCPERROR

Procedure name
tcerror

EZY4112E Too many levels of symbolic links (ELOOP)

Explanation
A pointer for a socket call parameter has too many levels of symbolic links for TCP to handle the call.

System action
The socket call ends. TCPIP continues.

Operator response
None.

System programmer response
Redefine the pointer for the socket call parameter in such a way that it has fewer levels of symbolic links.

Module
TCPERROR

Procedure name
tcerror

EZY4113E File name too long (ENAMETOOLONG)

Explanation
The data set name pointed to by the socket call is too long to be handled by TCPIP. The maximum number of characters is 248.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Change the pointer for the socket call parameter so that the data set pointed to has a shorter name.

Module
TCPERROR

Procedure name
tcerror

EZY4114E Host is down (EHOSTDOWN)
**Explanation**  
The host machine is down.

**System action**  
The socket call abends. TCPIP continues.

**Operator response**  
None.

**System programmer response**  
Try the call again after the host restarts.

**Module**  
TCPERROR

**Procedure name**  
tcerror

---

**EZY4115E**  
Host is unreachable (EHOSTUNREACH)

**Explanation**  
A connection to the host machine cannot be established.

**System action**  
The socket call abends. TCPIP continues.

**Operator response**  
None.

**System programmer response**  
Ensure that there is a network path to the specified host and that the host name is valid. If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. The traces will provide more information about the internal operations of socket functions and might suggest a reason for the error, or yield some better insight into the chronology of events leading up to the error. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**  
TCPERROR

**Procedure name**  
tcerror

---

**EZY4116E**  
Directory not empty (ENOTEMPTY)

**Explanation**  
The caller attempted to remove a directory that was not empty.
System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
None.

Module
TCPERROR

Procedure name
tcerror

EZY4117E Too many processes (EPROCLIM)

Explanation
Too many processes are running on the system.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Rewrite the socket call so as to close some of the active processes. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR

Procedure name
tcerror

EZY4118E Too many users (EUSERS)

Explanation
Too many users are on the system.

System action
The socket call abends. TCPIP continues.

Operator response
None.
System programmer response
Try the socket call again.

Module
TCPERROR

Procedure name
tcperror

EZY4119E Disc quota exceeded (EDQUOT)

Explanation
The client’s disk quota on the server has been exceeded.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Change the len parameter of the write() call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

Module
TCPERROR

Procedure name
tcperror

EZY4120E Stale NFS file handle (ESTALE)

Explanation
The Network File System (NFS) is unable to mount the MVS data set because the file handle that points to the data set no longer exists, or access to it has been revoked. Under NFS, each data set is referenced by the client through a file handle, which is a 32-bit identifier field.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Use the MOUNT command to update the data set handle. For more information about NFS, see z/OS Network File System Guide and Reference.
**Module**
TCPERROR

**Procedure name**
tcperror

**EZY4121E** Too many levels of remote in path (EREMOTE)

**Explanation**
The address structure pointed to by the socket call has too many levels of remote qualifiers.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Modify the address or name parameter for the socket call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZY4122E** Not a stream device (ENOSTR)

**Explanation**
A stream socket type was declared in the socket() call, but the remote device cannot transfer stream data.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Change the socket type to SOCK_DGRAM in the socket() call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZY4123E** Timer expired (ETIME)
**Explanation**
The timer for the socket call expired. The maximum interval to wait for completion was exceeded.

**System action**
The socket call ends. TCPIP continues.

**Operator response**
None.

**System programmer response**
If this message recurs, modify the timeout parameter of the select() call or change it to a NULL pointer. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZY4124E Out of stream resources (ENOSR)**

**Explanation**
No stream sockets are available to service the socket call.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Try the client application again.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZY4125E No message of desired type (ENOMSG)**

**Explanation**
There is no message of the correct data type for the socket.

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

System programmer response
None.

Module
TCPERROR

Procedure name
tcperror

EZY4126E Not a data message (EBADMSG)

Explanation
The message is a system or network control message.

System action
TCPIP continues.

Operator response
None.

System programmer response
None.

Module
TCPERROR

Procedure name
tcperror

EZY4127E Identifier removed (EIDRM)

Explanation
The access identifier for the file or data set has been removed.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
Use the mkpriv command to define the user privilege for the data set.
**Module**
TCPERROR

**Procedure name**
tcperror

**EZY4128E  Deadlock situation detected/avoided (EDEADLK)**

**Explanation**
A deadlock, the condition that occurs when multiple processes are waiting for the availability of a resource that will not become available because it is held by another process that is in a similar wait state, was detected.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. Review the trace output to determine why the deadlock occurred. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**
TCPERROR

**Procedure name**
tcperror

**EZY4129E  No record locks available (ENOLCK)**

**Explanation**
The host system does not provide for the locking of data sets.

**System action**
The socket call continues.

**Operator response**
None.

**System programmer response**
None.
**Procedure name**
tcerror

**EZY4130E**  Machine is not on the network (ENONET)

**Explanation**
The machine specified in the socket call is not on the host network.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Verify the address information in the socket call and modify as necessary.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY4131E**  Object is remote (ERREMOTE)

**Explanation**
The data set or disk specified is not on the local network.

**System action**
The socket call abends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Correct the address pointer in the socket call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.
If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY4132E**  The link has been severed (ENOLINK)
**Explanation**  
The physical connection linking the sockets has been broken at some point.

**System action**  
The socket call abends. TCPIP continues.

**Operator response**  
None.

**System programmer response**  
Try the socket call again.

**Module**  
TCPERROR

**Procedure name**  
tcerror

---

**EZY4133E Advertise error (EADV)**

**Explanation**  
The path specified in the adv (advertise) or share command is incorrect, and the data set, file, or disk cannot be accessed.

**System action**  
The socket call abends. TCPIP continues.

**Operator response**  
None.

**System programmer response**  
Verify the correct resource name and pathname. Reinitiate the socket call.

**Module**  
TCPERROR

**Procedure name**  
tcerror

---

**EZY4134E srmount error (ESRMNT)**

**Explanation**  
An error occurred when the Network File System (NFS) server attempted to place the remote file system in the user’s data set hierarchy.

**System action**  
The socket call abends. TCPIP continues.
Operator response
None.

System programmer response
Use the mount command (# mount) to mount the remote resource. When you mount a remote resource, it stays mounted only during the current session.

Module
TCPERROR

Procedure name
tcperror

EZY4135E Communication error on send (ECOMM)

Explanation
A communication error has occurred on a send() call.

System action
The socket call abends. TCPIP continues.

Operator response
None.

System programmer response
If the error can be reproduced, set trace socket to active in the TCPIP.DATA configuration file and rerun the application. Review the trace output to determine why the error occurred. If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror

EZY4136E Protocol error (EPROTO)

Explanation
A protocol error was detected for the socket call.

System action
The socket call abends. TCPIP continues.

Operator response
None.
**System programmer response**
Create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

**Module**
TCPERROR

**Procedure name**
tcerror

---

**EZY4137E  Multihop attempted (EMULTIHOP)**

**Explanation**
The socket call specified a multihop address link. That is, it specified a connection path that omitted some routing information.

**System action**
The socket call is bypassed.

**Operator response**
None.

**System programmer response**
Correct the address parameters for the socket call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

If you are still unable to determine the cause of the problem, create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

---

**Module**
TCPERROR

**Procedure name**
tcerror

---

**EZY4138E  Cross mount point (not an error) (EDOTDOT)**

**Explanation**
The file system is mounted from a remote machine.

**System action**
The socket call continues.

**Operator response**
None.

**System programmer response**
None.
Module
TCPERROR

Procedure name
tcperror

**EZY4139E** Remote address changed (EREMCHG)

**Explanation**
The remote address specified in the socket call has changed.

**System action**
The socket call ends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Verify the remote address and correct the address parameters of the socket call.

Module
TCPERROR

Procedure name
tcperror

**EZY4195E** Arg list too long (E2BIG)

**Explanation**
The argument list for the call is too long.

**System action**
The socket call fails. TCPIP continues.

**Operator response**
None.

**System programmer response**
Correct the arguments to the call. Reinitialize the application.

If you are still unable to determine the cause of the problem, re-create the problem with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror
**EZY4196E** Invalid argument (EINVAL)

**Explanation**
The argument specified to the socket call was not valid. The nature of the error depends on the particular call that triggers it. The following list displays some of the possible function calls that might return this error value:

<table>
<thead>
<tr>
<th>Call</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>accept</td>
<td>Listen was not called for this socket.</td>
</tr>
<tr>
<td>bind</td>
<td>The socket is already bound to an address.</td>
</tr>
<tr>
<td>connect</td>
<td>The specified name length is incorrect.</td>
</tr>
<tr>
<td>fcntl</td>
<td>Incorrect flags were specified.</td>
</tr>
<tr>
<td>givesocket</td>
<td>An incorrect client ID was entered.</td>
</tr>
<tr>
<td>ioctl</td>
<td>The request is incorrect or not supported.</td>
</tr>
<tr>
<td>select</td>
<td>One of the fields in the time-out structure is incorrect.</td>
</tr>
<tr>
<td>sendto</td>
<td>The target address length is incorrect for the specified address family.</td>
</tr>
<tr>
<td>shutdown</td>
<td>The shutdown condition is not 0, 1, or 2.</td>
</tr>
<tr>
<td>takesocket</td>
<td>The specified client ID is incorrect.</td>
</tr>
</tbody>
</table>

**System action**
The socket call ends. TCPIP continues.

**Operator response**
None.

**System programmer response**
Use the information provided in the above table as a guide for the possible conditions under which this error value can occur. Apply this to your application and, correct the parameters of the appropriate socket call. For more information, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

**Module**
TCPERROR

**Procedure name**
tcerror

**EZY4197E** Bad file number (EBADF)

**Explanation**
An invalid argument to the socket call was specified. The nature of the error depends on the particular call that triggers it.

**System action**
The socket call fails. TCPIP continues.
Operator response
None.

System programmer response
Correct the parameters of the appropriate socket call. See the z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference for information about System Error Return Codes.

If you are still unable to determine the cause of the problem, re-create the problem with CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperro

EZY4198E Last IUCV call (IUCV call) returned error number

Explanation
This message displays the error number returned from the IUCV call. It is usually followed by a message in the range of EZY3950X–EZY3978X that explains the IUCV error.

System action
The system action depends on the nature of the message that follows this one. For more information, see that specific message number.

Operator response
None.

System programmer response
The system action depends on the nature of the message that follows this one. For more information, see that specific message number.

Module
TCPERROR

Procedure name
tcperro

EZY4199E Unknown errno value=error number

Explanation
The error number displayed is not recognized as one of the values used by TCPIP to report an error condition. For more information about system error codes, see z/OS Communications Server: IP Sockets Application Programming Interface Guide and Reference.

System action
TCPIP continues.
Operator response
None.

System programmer response
Create the problem again with the CTRACE option SOCKET active. Contact the IBM Software Support Center with the original message and the traces.

Module
TCPERROR

Procedure name
tcperror

EZY4200E CallTransform ConvType Error : conversion_type

Explanation
The conversion type found does not match supported values (1–8). The supported values are:

<table>
<thead>
<tr>
<th>Value</th>
<th>Conversion Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SJISKANJI</td>
</tr>
<tr>
<td>2</td>
<td>JIS83KS</td>
</tr>
<tr>
<td>3</td>
<td>JIS78KJ</td>
</tr>
<tr>
<td>4</td>
<td>DECKANJI</td>
</tr>
<tr>
<td>5</td>
<td>EUCKANJI</td>
</tr>
<tr>
<td>6</td>
<td>KSC5601</td>
</tr>
<tr>
<td>7</td>
<td>HANGUEL</td>
</tr>
<tr>
<td>8</td>
<td>TCHINESE</td>
</tr>
</tbody>
</table>

System action
Control is returned to the TNSTMAS routine, where the session is terminated.

Operator response
Restart the session, specifying a valid conversion type.

System programmer response
None.

Module
TNDTRSVB
**EZY4201E CallTransform Verb Error. Verb: Verb**

**Explanation**
The verb specified in the configuration data set does not match the supported options. The supported options are: TERMINIT, CHEKTYPE, ASKTTYPE, TTYHELLO, MAPTOASC, GRFTOMAP, MAPTOGRF, ASCTOMAP, BREAKKEY, TTLOGOFF, TTYGDBYE, SHUTDOWN, and SIMCHECK.

**System action**
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -4.

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

**System programmer response**
Correct the configuration data set to specify a proper verb from the list above. For more information on configuring the Telnet server, see z/OS Communications Server: IP Configuration Reference.

**Module**
TNDTRSVB

---

**EZY4202E CALLTRANSFORM Terminal Id Error. Id: Term_Id**

**Explanation**
The terminal ID was found to be outside expected range. It is either negative or greater than the maximum number of Telnet server connections. The return code is set to -5.

**System action**
Control is returned to the TNSTMAS routine, where the session is terminated.

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

**System programmer response**
Make sure that the terminal ID is properly specified. For more information, see z/OS Communications Server: IP Configuration Reference.

**Module**
TNDTRSVB

---

**EZY4203E TNDBCSTM - Telnet DBCS Transform Mode - ERRLOG DDname not found - TNDBCSE**
**Explanation**
The Telnet server was unable to allocate ddname TNDBCSER. This is where all error messages for TELNET 3270 DBCS transform are written.

**System action**
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -1.

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

**System programmer response**
Update the JCL or TCPIP procedure to include TNDBCSER.

**Module**
TNDTRSVB

**Procedure name**
Simcheck

**EZY4204E**
TNDBCSTM - Telnet DBCS Transform Mode - Config DDName not found - TNDBCSCN

**Explanation**
The Telnet server was unable to allocate ddname TNDBCSCN. This is the TELNET 3270 DBCS transform configuration data set.

**System action**
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -99.

**Operator response**
None.

**System programmer response**
Update the JCL or TCPIP procedure to include TNDBCSCN.

**Module**
TNDTRSVB

**Procedure name**
Simcheck

**EZY4206I**
IBM TCP/IP TELNET SERVER DBCS SERVICE START AT time ON date

**Explanation**
This message indicates the time and date at which the Telnet session started.

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

System programmer response
None.

Module
TNDTRSVB

Procedure name
TTYHELLO

EZY4207E  TTYHELLO CodeConversion error. rc : rc

Explanation
An error was encountered in the Code Conversion routine used to convert message EZY4206I during a call to the procedure TTYHELLO. The return code in this message indicates the offset in the current input buffer at which the error occurred. This message is written to the error log.

System action
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -2.

Operator response
None.

System programmer response
Check the input data to determine why this error occurred.

Module
TNDTRSVB

Procedure name
TTYHELLO

EZY4208E  TTYHELLO Terminal Id error. id : terminal_id

Explanation
This message is written to the error log. The terminal ID was found to be outside the expected range. It is either negative or greater than the maximum number of Telnet server connections.

System action
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -3.

Operator response
None.
System programmer response
Make sure that the terminal ID is properly specified. For more information, see z/OS Communications Server: IP Configuration Reference.

Module
TNDTRSVB

Procedure name
TTYHELLO

EZY4209E TTYHELLO portno empty.

Explanation
This message is written to the error log. There is no free port number. The Telnet server exits.

System action
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -4.

Operator response
None.

System programmer response
Make sure that the TCPIP configuration data set contains a valid PORT statement for the Telnet server. For more information, see z/OS Communications Server: IP Configuration Reference.

Module
TNDTRSVB

Procedure name
TTYHELLO

EZY4210I KEY-IN YOUR TERMINAL TYPE & CONVERSION TYPE:

Explanation
This message prompts the user to enter the terminal type being used and the conversion type wanted.

System action
Telnet continues.

Operator response
Enter your terminal type and conversion type.

System programmer response
None.

Module
TNDTRSVB
EZY4211E   ASKTTYPE CodeConversion error. rc : rc

Explanation
This message is written to the error log. A code conversion error occurred during a call to the procedure ASKTTYPE. The return code in this message displays the offset in the current input buffer at which the error occurred.

System action
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -2.

Operator response
None.

System programmer response
Check the input data to determine why this error occurred, and respond as indicated.

Module
TNDTRSVB

Procedure name
ASKTTYPE

EZY4212E   ASKTTYPE portno not found. portno : port_number

Explanation
This message is written to the error log. A free port number was not found.

System action
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -1.

Operator response
Notify the system programmer of the error.

System programmer response
Make sure that the TCPIP configuration data set contains a valid PORT statement for the Telnet server. For more information, see the z/OS Communications Server: IP Configuration Reference.

Module
TNDTRSVB

Procedure name
ASKTTYPE

EZY4213E   ASKTTYPE Terminal Id error. Id : terminal_ID
Explanation
This message is written to the error log. The terminal ID was outside the expected range. It is either negative or
greater than the maximum number of Telnet server connections.

System action
Control is returned to the TNSTMAS routine, where the session is terminated. Return code is set to -3.

Operator response
Notify the system programmer of the error.

System programmer response
Make sure that the terminal identifier is properly specified in the TCPIP configuration data set. For more
information, see z/OS Communications Server: IP Configuration Reference.

Module
TNDTRSVB

Procedure name
ASKTTYPE

EZY4214E  UNKNOWN TERMINAL TYPE.

Explanation
The user specified an unsupported terminal type. The supported options are TTY, VT100, and VT282

System action
No Telnet session is established. TCPIP continues.

Operator response
Enter a valid terminal type.

System programmer response
None.

Module
TNDTRSVB

Procedure name
CHEKTYPE

EZY4215E  CHEKTYPE CodeConversion error. rc : rc

Explanation
A code conversion error occurred during a call to the procedure CHEKTYPE. The return code value displayed in
this message indicates the offset in the current input buffer at which the error occurred.

System action
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -2.
Operator response
None.

System programmer response
Check the input data to determine why this error occurred.

Module
TNDTRSVB

Procedure name
CHEKTYPE

**EZY4216E  UNKNOWN CONVERSION TYPE.**

Explanation
The user entered an unsupported conversion type. Supported values are: SFJ, J8N J8O, DEC, EUC, KSH, HAN, and TCH.

System action
No Telnet session is established. TCPIP continues.

Operator response
Enter a valid conversion type.

System programmer response
None.

Module
TNDTRSVB

Procedure name
CHEKTYPE

**EZY4217E  UNKNOWN TERMINAL & CONVERSION TYPE.**

Explanation
The user entered an unsupported terminal type and an unsupported conversion type. Supported Conversion types are SFJ, J8N J8O, DEC, EUC, KSH, HAN, and TCH. Supported Terminal types are TTY, VT100, or VT282.

System action
No Telnet session is established. Telnet continues.

Operator response
Enter a valid terminal type and valid conversion type.

Module
TNDTRSVB
CHEKTYPE Terminal Id error. Id : terminal_ID

Explanation
The terminal ID was outside expected range. It is either negative or greater than the maximum number of Telnet server connections.

System action
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -3.

Operator response
Notify the system programmer of the error.

System programmer response
Make sure that the terminal identifier is properly specified in the TCPIP configuration data set. For more information, see z/OS Communications Server: IP Configuration Reference.

Module
TNDTRSVB

TERMINIT data allocation error.

Explanation
A nonzero return code was received from the subroutine DtAlloc, which is used to allocate storage for Telnet data. An attempt was made to allocate a data buffer the size of SIMDATASIZE for conversion codes. The return code is set to -5.

System action
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -5.

Operator response
Notify the system programmer of the error.

System programmer response
Make sure that enough storage is available to Telnet to allocate this data set.

Module
TNDTRSVB

TERMINIT address table error. data : address_table
**Explanation**  
The address table is expected to be 0 prior to allocation. This is not the case. Storage might not have been freed.

**System action**  
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -6.

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

**System programmer response**  
Flush the address table and restart Telnet.

**Module**  
TNDTRSVB

**Procedure name**  
TERMINIT

---

**EZY4221E TERMINIT Terminal Id error. Id : terminal_ID**

**Explanation**  
The terminal ID was outside the expected range. It is either negative or greater than the maximum number of Telnet server connections.

**System action**  
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -3.

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

**System programmer response**  
Make sure that the terminal identifier is properly specified in the TCPIP configuration data set. For more information, see z/OS Communications Server: IP Configuration Reference.

**Module**  
TNDTRSVB

**Procedure name**  
TERMINIT

---

**EZY4223E MAPTOGRF Terminal Id error. Id : terminal_ID**

**Explanation**  
The terminal ID was outside the expected range. It is either negative or greater than the maximum number of Telnet server connections.

**System action**  
Control is returned to the TNSTMAS routine, where the session is terminated. The return code is set to -3.
Operator response
Notify the system programmer of the error.

System programmer response
Make sure that the terminal identifier is properly specified in the TCPIP configuration data set. For more information, see the z/OS Communications Server: IP Configuration Reference.

Module
TNDTRSVB

Procedure name
MAPTOGRF

EZY4225I +++ MAPTOGRF +++

EZY4226E Abnormal data in TransfTermFlag

Explanation
This message is written to the error log. The variable TransfTermFlag is not an expected value. The following are expected values: READPAT, READMODA, READMOD, READBUFF.

System action
TransfTermBufl is reset to zero. Processing continues.

Operator response
None.

System programmer response
None.

Module
TNDTRSVB

Procedure name
MAPTOGRF

EZY4225I +++ MAPTOGRF +++

EZY4227I TransfTermFlag : TransfTerm Flag

EZY4228I Output Data Length : TransfTerm Buffer length

EZY4229I Output Data Term Buffer

Explanation
These messages are written to the error log. These messages display internal trace information.

System action
Telnet continues.
Operator response
None.

System programmer response
None.

Module
TNDTRSVB

Procedure name
MAPTOGRF

**EZY4230I  +++ GRFTOMAP +++ RCV DATA LENGTH = length**

Explanation
The function GRFTOMAP, which is used to map conversion characters, is starting. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDTRSVB

Procedure name
GRFTOMAP

**EZY4231I  +++ GRFTOMAP +++ RCV DATA TBUFF**

Explanation
This message is displayed if the TRACE option is specified. This message displays trace information from the function GRFTOMAP, which is used to map conversion characters. Valid values are:

- EW, EWA --> Erase/Write order
- WRT --> Write order
- EAU --> Erase unprotected area
- RB --> Read Buffer
- RM --> Read modify
- RMA --> Read modify all
- WSF --> Write structured field
System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDTRSVB

Procedure name
GRFTOMAP

EZY4232I    EWorder EXEC LENGTH = Dleng

Explanation
This message occurs during processing of the ERASE/WRITE internal 3270 order. If Dleng is negative, sense code 08002 is set, indicating an ERASE/WRITE or ERASE/WRITE alternate command. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
Fix the user program and try it again.

Module
TNDTRSVB

Procedure name
GRFTOMAP

EZY4233I    WRTorder EXEC LENGTH = Dleng

Explanation
This message occurs during processing of the WRITE internal 3270 order. If Dleng is negative, sense code 08003 is set, indicating insufficient data in the WRITE command. This message is displayed if the TRACE option is specified.

System action
Telnet continues.
Operator response
None.

System programmer response
Fix the user program and try it again.

Module
TNDTRSVB

Procedure name
GRFTOMAP

EZY4234I EAUorder EXEC

Explanation
This message occurs during processing of the ERASE internal 3270 order. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDTRSVB

Procedure name
GRFTOMAP

EZY4235I RBorder EXEC

Explanation
This message occurs during processing of the Read Buffer internal 3270 order. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.
Module
TNDTRSVB

Procedure name
GRFTOMAP

**EZY4236I**  RMorder EXEC

**Explanation**
This message occurs during processing of the Read Modify internal 3270 order. This message is displayed if the TRACE option is specified.

**System action**
Telnet continues.

**Operator response**
None.

**System programmer response**
None.

Module
TNDTRSVB

Procedure name
GRFTOMAP

**EZY4237I**  RMAorder EXEC

**Explanation**
This message occurs during processing of the Read Modify All internal 3270 order. This message is displayed if the TRACE option is specified.

**System action**
Telnet continues.

**Operator response**
None.

**System programmer response**
None.

Module
TNDTRSVB

Procedure name
GRFTOMAP

**EZY4238I**  WSForder EXEC LENGTH = length
**Explanation**
This message occurs during processing of the Write Structured Field internal 3270 order. This message is displayed if the TRACE option is specified.

**System action**
Telnet continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
TNDTRSVB

**Procedure name**
GRFTOMAP

**EZY4240E** ++ GRFTOMAP ABNORMAL END !! ++ : SCode : NAddr SenseCode, ComBuffer Buffer Address ComBuffer.DataBuffer ComBuffer.AttrBuffer

**Explanation**
An error was encountered in the function GRFTOMAP, which is used to map conversion characters. The return code is set to -1.

**System action**
Telnet continues.

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

**System programmer response**
Use the sense code displayed in this message and the list of sense codes in the z/OS Communications Server: IP User's Guide and Commands to determine the cause of the error, and respond as indicated.

**Module**
TNDTRSVB

**Procedure name**
GRFTOMAP

**EZY4242E** TELNET SERVER ERROR. SENSE CODE : SenseCode

**Explanation**
An error has been found during processing. The associated Sense code is printed. A list of the sense codes can be found in the z/OS Communications Server: IP User's Guide and Commands.
System action
Telnet continues.

Operator response
Notify the system programmer of the error.

System programmer response
Use the sense code displayed in this message and the list of sense codes in the z/OS Communications Server: IP User's Guide and Commands to determine the cause of the error and respond as indicated.

Module
TNDTRSVB

Procedure name
TTYGDBYE

EZY4243I  IBM TCP/IP TELNET SERVER DBCS SERVICE END AT time ON date

Explanation
This message displays the time and date at which the Telnet session ended.

System action
Telnet ends normally. TCPIP continues.

Operator response
None.

System programmer response
None.

Module
TNDTRSVB

Procedure name
TTYGDBYE

EZY4244E  TTYGDBYE CodeConversion error. rc: rc

Explanation
A code conversion error occurred during a call to the procedure TTYGDBYE. The return code value displayed in this message indicates the offset in the current input buffer at which the error occurred.

System action
Control is returned to the calling routine TNSTMAS and the session is terminated. The return code is set to -2.

Operator response
None.
System programmer response
Check the input data to determine why the error occurred.

Module
TNDTRSVB

Procedure name
TTYGDBYE

EZY4245E  TTYGDBYE portno not found. portno : port number

Explanation
The indicated port number was not found.

System action
Control is returned to the calling routine TNSTMAS and the session is terminated. The return code is set to -1.

Operator response
Reissue the TELNET command specifying a valid port number. For more information about the TELNET command, see the z/OS Communications Server: IP User's Guide and Commands.

System programmer response
If the error persists, make sure that the TCPIP configuration data set for the client contains a valid PORT statement for the Telnet server. For more information, see the z/OS Communications Server: IP Configuration Reference.

Module
TNDTRSVB

Procedure name
TTYGDBYE

EZY4246E  TTYGDBYE Terminal Id error. id : terminal ID

Explanation
The terminal ID was outside the expected range. It is either negative or greater than the maximum number of Telnet server connections.

System action
Control is returned to the calling routine TNSTMAS and the session is terminated. The return code is set to -3.

Operator response
Notify the system programmer of the error.

System programmer response
Make sure that the terminal identifier is properly specified in the client's TCPIP configuration data set. For more information, see z/OS Communications Server: IP Configuration Reference.
EZY4300I  ***** WRTorder DATA LENGTH = length

Explanation
This message displays the length of data being passed to the function WRTorder, which is used to write data to the terminal. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WRTorder

EZY4301I  #SFsbudr EXEC

Explanation
This message displays trace information indicating that the user is currently in the Start Field branch of a case statement. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.
Explanation
This message displays trace information indicating that the user is currently in the Start Field Extent branch of a case statement. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WRTorder

EZY4303I

Explanation
The user is in the Modify Field branch of a case statement. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WRTorder

EZY4304I

Explanation
The user is in the Set Buffer Address branch of a case statement. This message is displayed if the TRACE option is specified.

System action
Telnet continues.
Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WRTorder

**EZY4305I**  ############### ICbuodr EXEC

**Explanation**
The user is in the Insert Cursor branch of a case statement. This message is displayed if the TRACE option is specified.

**System action**
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WRTorder

**EZY4306I**  ############### PTbuodr EXEC

**Explanation**
The user is in the Program Tab branch of a case statement. This message is displayed if the TRACE option is specified.

**System action**
Telnet continues.

Operator response
None.

System programmer response
None.
Module
TND3270

Procedure name
WRTorder

EZY4307I  #RRAsbuodr EXEC

Explanation
The user is in the Repeat Address branch of a case statement. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WRTorder

EZY4308I  #RRAsbuodr EXEC

Explanation
The user is in the Erase Unprotected to Address branch of a case statement. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WRTorder

EZY4309I  SA EXEC

472  z/OS Communications Server: IP Messages: Volume 3 (EZY)
Explanation
This message occurs during processing of the Set Attribute internal 3270 order. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WRTorder

EZY4310I WriteData EXEC LENGTH = length

Explanation
The user is in the Normal Character branch of a case statement. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WRTorder

EZY4311I IDcnt = ID count

Explanation
This message displays the value of IDcount, a counter used in processing of the Write Structured Field order. This message is displayed if the TRACE option is specified.

System action
Telnet continues.
Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WSForder

EZY4312I READ PARTITION

Explanation
A Read Partition request is being processed. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WSForder

EZY4313I ErstID EXEC i = i

Explanation
An Erase/Reset request is being processed. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.
Module
TND3270

Procedure name
WSForder

EZY4314I  SetRepID EXEC i = i

Explanation
The Telnet server is in set reply mode. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WSForder

EZY4315I  OutbDtID EXEC i = i

Explanation
This message indicates the amount of data being sent to a 3270-type terminal. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
WSForder

EZY4316I  OutbDtID ( WRTorder ) Exec Dataleng = Data length
Explanation
This message indicates the amount of data being passed to the function WRTorder, which is used to write data to a terminal. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
OutBdtID

EZY4317I  OutbDtID ( EWorder ) Exec Dataleng = Data length

Explanation
This message displays the amount of data being passed to the function EWorder. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TND3270

Procedure name
OutBdtID

EZY4318I  OutbDtID ( EAUorder ) EXEC

Explanation
This message indicates the function EAUorder is being called. This message is displayed if the TRACE option is specified.

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

**System programmer response**
None.

**Module**
TND3270

**Procedure name**
OutBdtID

**EZY4319I**  WRTorder error found with data procedure_name

**Explanation**
The DBCS transform function found incorrect data in the data stream. The buffer containing the incorrect data could not be transformed.

In the message text:

*procedure_name*

  The name of the procedure where the incorrect data was found.

**System action**
The transform of this buffer stops. The DBCS transform function continues with the next buffer.

**Operator response**
Correct the error and try the transform again.

**System programmer response**
None.

**Module**
TND3270

**Procedure name**
EWorder, OUTbDtID

**EZY4350I**  +++ BuffShift (Buffer Shift) +++

**EZY4351E**  First Parameter is Out of Range

**Explanation**
A request was made to shift data in a buffer, but the address passed is out of range. This message is written to the error log.

**System action**
No data is moved. Control is returned to the calling routine and processing continues.
Operator response
None.

System programmer response
None.

Module
TNDMISCL

Procedure name
BUFFSHIFT

EZY4350I  +++ BuffShift (Buffer Shift) +++
EZY4352I  Second Parameter is Out of Range

Explanation
A request was made to shift data in a buffer, but the address passed is out of range. This message is written to the error log.

System action
No data is moved. Control is returned to the calling routine and processing continues.

Operator response
None.

System programmer response
None.

Module
TNDMISCL

Procedure name
BUFFSHIFT

EZY4350I  +++ BuffShift (Buffer Shift) +++
EZY4353I  Third Parameter is Out of Range

Explanation
A request was made to shift data in a buffer, but the address passed is out of range.

System action
No data is moved. Control is returned to the calling routine and processing continues.

Operator response
None.
System programmer response
None.

Module
TNDMISCL

Procedure name
BUFFSHIFT

EZY4354W +++ OpenCodeTable : warning +++
EZY4356I ConType= Con Type ---> IBM *

Explanation
Either no conversion type was specified or an incorrect conversion type was found. Valid options are: J7O, J7N, J8O, J8N, A7O, A7N, A8O, A8N, SFJ, DEC, EUC, IBM, KSH, HAN, and TCH. This message is written to the error log.

System action
The conversion type is set to IBM, and Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDMISCL

Procedure name
OpenCodeTable

EZY4357W +++ CodeConversion : warning +++
EZY4358I ConDir= Con Dir ---> 0 *

Explanation
An unexpected value was found for ConDir.

System action
The value is reset to 0, and Telnet continues.

Operator response
None.

System programmer response
None.
Module
TNDMISCL

Procedure name
CodeConversion

EZY4357W  +++ CodeConversion : warning +++
EZY4359I  InState= InState ---> 0 *

Explanation
An unexpected value was found for InState.

System action
The value is reset to 0, and Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDMISCL

Procedure name
CodeConversion

EZY4357W  +++ CodeConversion : warning +++
EZY4360I  OutState= OutState ---> 0 *

Explanation
An unexpected value was found for OutState.

System action
The value is reset to 0, and Telnet continues.

Operator response
None.

System programmer response
None.
EZY4357W  +++ CodeConversion : warning +++

EZY4361I  ConMode= ConMode ---> 0 *

Explanation
An unexpected value was found for ConMode.

System action
The value is reset to 0, and Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDMISCL

Procedure name
CodeConversion

EZY4362W  +++ CloseCodeTable : warning +++

EZY4356I  ConType= Con Type ---> IBM *

Explanation
Either no conversion type was specified or an incorrect conversion type was found. Valid options are: J7O, J7N, J8O, J8N, A7O, A7N, A8O, A8N, SFJ, DEC, EUC, IBM, KSH, HAN, and TCH.

System action
The conversion type is set to IBM, and Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDMISCL

Procedure name
CloseCodeTable

EZY4363I  *CCTCLOS*

EZY4364I  ists= ists
Explanation
This message displays trace information. This message is displayed if the TRACE option is specified.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDMISCL

Procedure name
CloseCodeTable.

EZY4366E  TNDBCSTM - Telnet DBCS Transform Mode - CodeFile DDName not found - DD name

Explanation
The Telnet server was unable to allocate ddname TNDBCSXL. This is the Telnet 3270 DBCS Transform data set containing binary translation table code files. This message is written to the error log.

System action
Telnet continues.

Operator response
None.

System programmer response
Update the JCL or TCPIP procedure to include TNDBCSTM.

Module
TNDMISCL

Procedure name
RdCodeFiles

EZY4380I  +++ ASCII TO MAP (ATMDEC) +++
EZY4381I  Input Data Length : Terminal Buffer length
EZY4382I  Input Data Term Buffer

Explanation
This message displays trace information about the TERMBUF buffer, which contains data mapping the terminal keys to hexadecimal characters. All of these key comparisons are declared as constants in the TNDCOMM macro. These messages are displayed if the TRACE option is specified.
System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDATMAP

Procedure name
ATMDEC

EZY4400I ****** MAP to ASCII ******
EZY4401I TransfTermBuff : Set Data Length Over
EZY4402E Error Position No. is sequence number
EZY4403I Data Length is length

Explanation
An error was found after calling the internal routine CheckBufl. The sequence number specified corresponds to the position in the routine where the failure occurred. These messages are written to the error log.

System action
Telnet ends normally. TCPIP continues.

Operator response
None.

System programmer response
Save all trace information for problem determination and contact the IBM Software Support Center.

Module
TNDMTASC

Procedure name
SetERRlog

EZY4400I ****** MAP to ASCII ******
EZY4404I Kanji Code Conversion : rt = rc

Explanation
An error occurred during code conversion while mapping input data to ASCII characters. The return code value displayed in this message indicates the offset in the current input buffer at which the error occurred. These messages are written to the error log.
System action
Telnet continues.

Operator response
None.

System programmer response
Check the input data to determine why this error occurred. and respond as indicated.

Module
TNDMTASC

Procedure name
CharConv

EZY4400I ****** MAP to ASCII ******
EZY4405I Alpha Code Conversion : rt = rc

Explanation
An error occurred during code conversion while mapping input data to ASCII characters. The return code value displayed in this message indicates the offset in the current input buffer at which the error occurred. These messages are written to the error log.

System action
Telnet continues.

Operator response
None.

System programmer response
Check the input data to determine why this error occurred.

Module
TNDMTASC

Procedure name
CharConv

EZY4400I ****** MAP to ASCII ******
EZY4406E Compare Error Address : Cradr
EZY4407I WritBuffer Data : Dump of DataBuffer
EZY4408I AttrBuffer Data : Dump of AttrBuffer
EZY4409I Disp.WritBuffer Data : Dump of DispBuffer.DataBuffer
EZY4410I Disp.AttrBuffer Data : Dump of DispBuffer.Attrbuffer
Explanation
These messages display trace information from the CodeConversion function. These messages are written to the error log.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDMTASC

Procedure name
DTCOMP

**EZY4400I** ******* MAP to ASCII *******

**EZY4411I**  Alarm Code Conversion : rt = rc

Explanation
An error occurred during code conversion while mapping input data to ASCII characters. The return code value displayed in this message indicates the offset in the current input buffer at which the error occurred.

System action
Telnet continues.

Operator response
Notify the system programmer.

System programmer response
Check the input data to determine why this error occurred.

Module
TNDMTASC

Procedure name
MTADEC

**EZY4400I** ******* MAP to ASCII *******

**EZY4412I**  Output Data Length : length

**EZY4413I**  Output Data dump of TermBuff
Explanation
These messages display information about the contents of TermBuff, which is used to hold information mapping
terminal keys to hexadecimal characters. These messages are written to the error log.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDMTASC

Procedure name
MTADEC

**EZY4414I**  DataBuffer dump of DataBuffer

**EZY4415I**  AttrBuffer dump of AttrBuffer

Explanation
Displayed is the value of the data and attribute buffers. These messages display during trace and are written to
the error log.

System action
Telnet continues.

Operator response
None.

System programmer response
None.

Module
TNDMTASC

Procedure name
MTADEC

**EZY4450I**  +++ CursorUp +++

**EZY4451E**  Cursor Address is Out of Range. Non Process End.

Explanation
The placement of the cursor exceeded the supported buffer size of 1920. These messages are written to the
error log.
System action
The process ends normally.

Operator response
None.

System programmer response
Make sure that the terminal type is properly specified and that the terminal is supported by Telnet.

Module
TNDTERM

Procedure name
CursorUp

EZY4453I  +++ CursorDown +++
EZY4451E  Cursor Address is Out of Range. Non Process End.

Explanation
The placement of the cursor exceeded the supported buffer size of 1920. These messages are written to the error log.

System action
The process ends normally.

Operator response
None.

System programmer response
Make sure that the terminal type is properly specified and that the terminal is supported by Telnet.

Module
TNDTERM

Procedure name
CursorDown

EZY4454I  +++ CursorLeft +++
EZY4451E  Cursor Address is Out of Range. Non Process End.

Explanation
The cursor placement exceeded the supported size of 1920 bytes. These messages are written to the error log.

System action
The process ends normally.
Operator response
None.

System programmer response
Make sure that the terminal type was properly specified and that the terminal is supported by Telnet.

Module
TNDTERM

Procedure name
CursorLeft

EZY4455I +++ CursorRight +++
EZY4451E Cursor Address is Out of Range. Non Process End.

Explanation
The placement of the cursor exceeded the supported length of 1920.

System action
The process ends normally.

Operator response
None.

System programmer response
Make sure that the terminal type is properly specified and that the terminal is supported by Telnet.

Module
TNDTERM

Procedure name
CursorRight

EZY4456E +++ CursHome +++ Home Position Address is Out of Range. Home Position is set to 0 for Next process.

Explanation
The cursor is not found at the home position as expected. This message is written to the error log.

System action
The cursor is reset to 0 and processing continues.

Operator response
Reset the cursor position if necessary.

System programmer response
None.
Module
TNDTERM

Procedure name
CursHome

EZY4459E  +++ Reset Key +++
EZY4451E  Cursor Address is Out of Range. Abnormal end.

Explanation
The cursor was placed out of the supported range of 1920. This message is written to the error log.

System action
Processing continues, the return code is set to 1.

Operator response
Reposition the cursor within the accepted range of 0 to 1920.

System programmer response
None.

Module
TNDTERM

Procedure name
ResetKey

EZY4461I  +++ FieldMark +++
EZY4451E  Cursor Address is Out of Range.

Explanation
The cursor was placed beyond the supported range of 1920.

System action
Processing continues.

Operator response
Reposition the cursor within the accepted range of 0 to 1920.

System programmer response
None.
EZY4461I  +++ FieldMark +++

EZY4463E  Attribute Address is Out of Range.

Explanation
The given attribute address is out of the valid range.

System action
Control is returned to the calling routine and processing continues.

Operator response
Reenter the command specifying a valid attribute.

System programmer response
None.

Module
TNDTERM

Procedure name
FieldMark

EZY4461I  +++ FieldMark +++
EZY4464I  Formatted Buffer.
EZY4472E  Attribute Undefined.

Explanation
The function SearchAttr, which is used to check conversion attributes, encountered an undefined attribute.

System action
Control is returned to the calling routine.

Operator response
Resubmit the last request specifying valid attributes. For more information, see the z/OS Communications Server: IP User’s Guide and Commands.

System programmer response
None.

Module
TNDTERM

Procedure name
FieldMark

EZY4465I  +++ FieldTab +++
EZY4451E  Cursor Address is Out of Range. Address is set to 0 for normal end.
Explanation
The cursor was placed outside the valid range of 0 to 1920.

System action
The cursor address is set to 0 and processing continues.

Operator response
Reposition the cursor if necessary.

System programmer response
None.

Module
TNDTERM

Procedure name
FieldTab

EZY4468I +++ FieldBackTab +++
EZY4451E Cursor Address is Out of Range. Address is set to 0 for normal end.

Explanation
The cursor was placed outside the supported range of 0 to 1920.

System action
The cursor address is set to 0 and processing continues.

Operator response
Reposition the cursor if necessary.

System programmer response
None.

Module
TNDTERM

Procedure name
FieldBackTab

EZY4469I +++ EraseEOF +++
EZY4451E Cursor Address is Out of Range.

Explanation
The cursor was placed outside the supported range of 0 to 1920.

System action
Processing continues.
Operator response
Reposition the cursor if necessary.

System programmer response
None.

Module
TNDTERM

Procedure name
EraseEOF

EZY4469I  +++ EraseEOF +++
EZY4464I  Formatted Buffer.
EZY4463E  Attribute Address is Out of Range.

Explanation
The function SearchAttr, which is used to check conversion attributes, encountered an incorrect attribute.

System action
The request is not processed. Telnet continues.

Operator response
Resubmit the request, specifying a valid attribute.

System programmer response
None.

Module
TNDTERM

Procedure name
EraseEOF

EZY4469I  +++ EraseEOF +++
EZY4464I  Formatted Buffer.
EZY4472E  Attribute Undefined.

Explanation
The function SearchAttr, which is used to check conversion attributes, encountered an undefined attribute.

System action
The request is not processed. Telnet continues.

Operator response
Resubmit the request specifying valid attributes.
System programmer response
None.

Module
TNDTERM

Procedure name
EraseEOF

EZY4473I +++ EraseInput +++
EZY4464I Formatted Buffer.
EZY4472E Attribute Undefined.

Explanation
The function SearchAttr, which is used to check conversion attributes, encountered an undefined attribute.

System action
The request is not processed. Telnet continues.

Operator response
Resubmit the request specifying valid attributes.

System programmer response
None.

Module
TNDTERM

Procedure name
EraseInput

EZY4474E +++ AttPAKey +++ PA key ASCII Code Not found

Explanation
The PA key used is unsupported. Supported values are:
- PA1 (hex 6C for Ebcidic hex 31 for ASCII)
- PA2 (hex 6E for Ebcidic hex 32 for ASCII)
- PA3 (hex 6B for Ebcidic hex 33 for ASCII)

All other keys are ignored. The return code is set to 1.

System action
The system returns to caller and continues processing.

Operator response
Specify the correct PA key.
System programmer response
None.

Module
TNDTERM

Procedure name
AttPAKey

EZY4476I +++ AttPFKey +++
EZY4477E PF key No. out of Range

Explanation
The PF key used is unsupported. Supported values are PF1—PF24. The return code is set to 1.

System action
Control returns to the caller and processing continues.

Operator response
Specify a valid PF key.

System programmer response
None.

Module
TNDTERM

Procedure name
AttPFKey

EZY4478I +++ NextCurs +++
EZY4451E Cursor Address is Out of Range.

Explanation
The cursor was placed outside the supported range of 0 to 1920.

System action
Processing continues.

Operator response
Reposition the cursor within the supported range.

System programmer response
None.
**Explanation**
The internal subroutine SearchAddr, which is used to check conversion attributes, encountered an undefined attribute.

**System action**
The request is not processed. Telnet continues.

**Operator response**
Resubmit the request specifying valid attributes.

**System programmer response**
None.

**Module**
TNDTERM

**Procedure name**
NextCurs

**Explanation**
The cursor was placed outside the supported range of 0 to 1920.

**System action**
Control is returned to the calling routine.

**Operator response**
Reposition the cursor within the supported range.

**System programmer response**
None.

**Module**
TNDTERM

**Procedure name**
MstReset
EZY4451E  Cursor Address is Out of Range.

Explanation
The cursor was placed outside of the supported range of 0 to 1920.

System action
Control is returned to the calling routine.

Operator response
None.

System programmer response
None.

Module
TNDTERM

Procedure name
DUPRIC

EZY4480I  +++ Dupric +++
EZY4464I  Formatted Buffer.
EZY4463E  Attribute Address is Out of Range.

Explanation
The internal subroutine SearchAttr, which is used to check conversion attributes, encountered an attribute that was out of range.

System action
The request is not processed. Telnet continues.

Operator response
Resubmit the request specifying valid attributes.

System programmer response
None.

Module
TNDTERM

Procedure name
DUPRIC

EZY4480I  +++ Dupric +++
EZY4464I  Formatted Buffer.
EZY4472E  Attribute Undefined.
**Explanation**
The internal subroutine SearchAttr, which is used to check conversion attributes, encountered an undefined attribute.

**System action**
The request is not processed. Telnet continues.

**Operator response**
Resubmit the request specifying valid attributes.

**System programmer response**
None.

**Module**
TNDTERM

**Procedure name**
DUPRIC

**EZY4481I**  +++ CrgReturn +++

**EZY4451E**  Cursor Address is Out of Range.

**Explanation**
The cursor was placed outside the supported range of 0 to 1920. The return code is set to 1.

**System action**
Control is returned to the calling routine.

**Operator response**
None.

**System programmer response**
None.

**Module**
TNDTERM

**Procedure name**
CrgReturn

**EZY4482I**  +++ NormalChar (Input) +++

**EZY4451E**  Cursor Address is Out of Range.

**Explanation**
The cursor was placed outside the supported range of 0 to 1920.
System action
Control returns to the calling routine.

Operator response
Reposition the cursor within the supported range.

System programmer response
None.

Module
TNDTERM

Procedure name
NormalChar

EZY4482I  +++ NormalChar (Input) +++
EZY4464I  Formatted Buffer.
EZY4463E  Attribute Address is Out of Range.

Explanation
The internal subroutine SearchAttr, which is used to check conversion attributes, encountered an out of range attribute.

System action
The request is not processed. Telnet continues.

Operator response
Resubmit the request specifying valid attributes.

System programmer response
None.

Module
TNDTERM

Procedure name
NormalChar

EZY4482I  +++ NormalChar (Input) +++
EZY4464I  Formatted Buffer.
EZY4472E  Attribute Undefined.

Explanation
The internal subroutine SearchAttr, which is used to check conversion attributes, encountered an undefined attribute.
**System action**  
The request is not processed. Telnet continues.

**Operator response**  
Resubmit the request specifying valid attributes.

**System programmer response**  
None.

**Module**  
TNDTERM

**Procedure name**  
NormalChar

<table>
<thead>
<tr>
<th>EZY4483I</th>
<th>+++ Delete1C +++</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZY4451E</td>
<td>Cursor Address is Out of Range.</td>
</tr>
</tbody>
</table>

**Explanation**  
The cursor was placed outside the supported range of 0 to 1920.

**System action**  
Control is returned to the calling routine.

**Operator response**  
Reposition the cursor within the valid range.

**System programmer response**  
None.

**Module**  
TNDTERM

**Procedure name**  
Delete1C

<table>
<thead>
<tr>
<th>EZY4483I</th>
<th>+++ Delete1C +++</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZY4464I</td>
<td>Formatted Buffer.</td>
</tr>
<tr>
<td>EZY4472E</td>
<td>Attribute Undefined.</td>
</tr>
</tbody>
</table>

**Explanation**  
The internal subroutine SearchAttr, which is used to check conversion attributes, encountered an undefined attribute.

**System action**  
The request is not processed. Telnet continues.
Operator response
Resubmit the request specifying valid attributes.

System programmer response
None.

Module
TNDTERM

Procedure name
Delete1C

EZY4483I  +++ Delete1C +++
EZY4464I  Formatted Buffer.
EZY4463E  Attribute Address is Out of Range.

Explanation
The internal subroutine SearchAttr, which is used to check conversion attributes, encountered an out of range attribute.

System action
The request is not processed. Telnet continues.

Operator response
Resubmit the request specifying valid attributes.

System programmer response
None.

Module
TNDTERM

Procedure name
Delete1C

EZY4484I  ++ Inputerr ++ message

Explanation
The Telnet DBCS code has detected an error with the input data. Some of the possible reasons are:
• The keyboard is locked
• Input was tried in a protected field
• The command was out of sequence
• The data entered was not valid.

An informational message is displayed in the variable text field of the message. The layout of the variable text field is described below:

xxxxxxxx-yyyyyy nnnnnnnnnnnnnnnnnnnn
where

**xxxxxx**
The translation routine being used. This routine can be correlated with the code page being used and the translation being done. The first few characters are the translation and the rest, if included, the code page being used.

<table>
<thead>
<tr>
<th>Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATM - ASCII to 3270</td>
</tr>
<tr>
<td>TNDTERM - 3270 terminal translation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Codepage</th>
</tr>
</thead>
<tbody>
<tr>
<td>SJF - SJISKANJI</td>
</tr>
<tr>
<td>J80 - JIS78KJ</td>
</tr>
<tr>
<td>J8N - JIS83KJ</td>
</tr>
<tr>
<td>EUC - EUCKANJI</td>
</tr>
<tr>
<td>DEC - DECKANJI</td>
</tr>
<tr>
<td>HAN - HANGEUL</td>
</tr>
<tr>
<td>KSC - KSC5601</td>
</tr>
<tr>
<td>TCH - TCHINESE</td>
</tr>
</tbody>
</table>

**yyyyy**
The command(s) being processed. There might be multiple commands strung together all separated by a hyphen (-). An example of this would be

```
NULL-ESC-CRTLK
```

*nnnnnn*
The reason.

Possible reasons and actions are:

<table>
<thead>
<tr>
<th>Reason</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ATMDEC-CtrlA INPUT INHIBITED</strong></td>
<td>The keyboard is locked. Reset the keyboard.</td>
</tr>
<tr>
<td><strong>ATMDEC-NULL INVALID COMMAND RECEIVED</strong></td>
<td>The previous command was not valid. Try again.</td>
</tr>
<tr>
<td>ATMDEC x</td>
<td>The command/data was either not valid or out of sequence.</td>
</tr>
</tbody>
</table>

**System action**
TCPIP continues.

**Operator response**
Resubmit the request specifying valid attributes.

**System programmer response**
From the trace message try to determine the cause of the error, if unable, then contact the IBM Software Support Center.

**Module**
TNDTERM TNDATMAP

**Procedure name**
various
Chapter 7. EZY5xxxx messages

EZY5398E  REQUEST TO START TCP/IP VIA MVPMAIN IS REJECTED.

Explanation
Program MVPMAIN was invoked with MODULE=TCPIP specified on the PARM= parameter of the JCL EXEC statement. The TCP/IP address space can no longer be started using program MVPMAIN.

System action
The task initialization function is terminated.

Operator response
Tell the system programmer about the error.

System programmer response
See sample procedure TCPIPROC in the SEZAINST data set for the correct program name to be used for starting the TCP/IP address space.

Module
MVPMAIN

Procedure name
Mainline code
Chapter 8. EZY6xxx messages

EZY6001I EZAZSSI Already Active, Start Rejected

Explanation
EZAZSSI is already active, attempting to start TNF and/or VMCF.

System action
This duplicate copy of EZAZSSI ends.

Operator response
Either wait for VMCF and/or TNF to start, cancel then restart EZAZSSI, or ignore the message.

System programmer response
None.

Module
EZAZSSI

EZY6002I TNF Already Active

Explanation
TNF is already active, therefore it is not started.

System action
Start VMCF if needed, then EZAZSSI ends.

Operator response
Either end TNF and restart, or ignore the message.

System programmer response
None.

Module
EZAZSSI

EZY6003I VMCF Already Active

Explanation
VMCF is already active, therefore it is not started.

System action
EZAZSSI ends.

Operator response
Either end VMCF and restart, or ignore the message.
**System programmer response**
None.

**Module**
EZAZSSI

**EZY6004I**  
*No Action Taken*

**Explanation**
VMCF and TNF are both running; there is nothing to start.

**System action**
TNF/VMCF continue, EZAZSSI ends.

**Operator response**
Either stop VMCF and/or TNF then restart EZAZSSI, or ignore the message.

**System programmer response**
None.

**Module**
EZAZSSI

**EZY6005I**  
*Invalid Start Sequence in Progress*

**Explanation**
VMCF is up and running and TNF is down.

**System action**
VMCF continues, EZAZSSI ends.

**Operator response**
Stop VMCF and then restart EZAZSSI.

**System programmer response**
None.

**Module**
EZAZSSI

**EZY6006E**  
*Subsystem name in IEFSSNxx is not VMCF*

**Explanation**
There is no VMCF entry in IEFSSNxx.

**System action**
EZAZSSI ends.
**Operator response**
None.

**System programmer response**
Add the VMCF entry to your IEFSSNxx parmlib member. Either load the initial program to incorporate this change or use the SETSSI ADD command to dynamically add the VMCF subsystem, and then restart EZAZSSI. See z/OS MVS System Commands for more information about the SETSSI command.

**Module**
EZAZSSI

**EZY6007E** VMCF Nodename must be specified

**Explanation**
VMCF requires that a nodename be specified at least once per IPL.

**System action**
EZAZSSI ends.

**Operator response**
Specify a nodename when starting EZAZSSI. See the z/OS Communications Server: IP Diagnosis Guide for more information.

**System programmer response**
None.

**Module**
EZAZSSI

**EZY6008I** VMCF Start Initiated

**Explanation**
A VMCF address space start was issued.

**System action**
EZAZSSI continues; start of VMCF expected.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAZSSI

**EZY6009E** Refused to create VMCF address space
**Explanation**
The VMCF address space start from IEEMB881 failed.

**System action**
EZAZSSI ends.

**Operator response**
None.

**System programmer response**
Check to see that the data sets containing the respective load modules are cataloged and available.

**Module**
EZAZSSI

**EZY6010E VMCF Not Initialized, Processing Continues**

**Explanation**
The VMCF address space start-up timed out. See the messages that usually accompany this message for an indication of the actual problem.

**System action**
EZAZSSI ends.

**Operator response**
Determine why VMCF did not start and restart EZAZSSI.

**System programmer response**
None.

**Module**
EZAZSSI

**EZY6011I VMCF Initialization Complete**

**Explanation**
VMCF is active and available. VMCF applications can be started.

**System action**
VMCF continues, EZAZSSI ends.

**Operator response**
None.

**System programmer response**
None.
Module
EZAZSSI

**EZY6012E**  Error creating PC numbers, is LPA correct?

**Explanation**
PC numbers could not be created.

**System action**
TNF/VMCF ends, EZAZSSI ends.

**Operator response**
None.

**System programmer response**
This is probably an installation error. Check the LPA data sets.

Module
EZAZSSI

**EZY6013E**  VMCF ended: Error in initialization

**Explanation**
An error was encountered in initialization. See the messages that usually accompany this message for an indication of the actual problem.

**System action**
VMCF ends.

**Operator response**
Dependent on accompanying messages.

**System programmer response**
Dependent on accompanying messages.

Module
EZAZSSI

**EZY6014E**  Subsystem name in IEFSSNxx is not TNF

**Explanation**
There is no TNF entry in IEFSSNxx.

**System action**
EZAZSSI ends.

**Operator response**
None.

EZY6xxx messages 509
System programmer response
Add the TNF entry to your IEFSSNxx parmlib member. Either IPL to incorporate this change or use the SETSSI ADD command to dynamically add the TNF subsystem, and then restart EZAZSSI. See z/OS MVS System Commands for more information about the SETSSI command.

Module
EZAZSSI

EZY6015I  TNF Start Initiated

Explanation
A TNF address space start was issued.

System action
EZAZSSI continues; start of TNF expected.

Operator response
None.

System programmer response
None.

Module
EZAZSSI

EZY6016E  Refused to create TNF address space

Explanation
The TNF address space start from IEEMB881 failed.

System action
EZAZSSI ends.

Operator response
None.

System programmer response
Check to see that the data sets containing the respective load modules are cataloged and available.

Module
EZAZSSI

EZY6017E  TNF Not Initialized, Processing Continues

Explanation
The TNF address space start-up timed out. See the messages that accompany this message for an indication of the actual problem.
System action
EZAZSSI ends.

Operator response
Determine why TNF did not start and restart EZAZSSI.

System programmer response
None.

Module
EZAZSSI

EZY6018I   TNF Initialization Complete

Explanation
TNF is active and available. TNF applications can be started.

System action
EZAZSSI will attempt to start VMCF.

Operator response
None.

System programmer response
None.

Module
EZAZSSI

EZY6019E   TNF ended: MVPTTRML Resource Manager Not Established

Explanation
A problem exists in the RESMGR interface.

System action
TNF ends, EZAZSSI ends.

Operator response
None.

System programmer response
This is probably an installation error. This message can be issued if resource manager load module MVPTTRML is not found in LPA. Ensure that the TCP/IP LPA load library SEZALPA is defined in one of the LPALSTxx members of SYS1.PARMLIB.

Module
EZAZSSI

EZY6020I   TNF not started correctly
Explanation
TNF was started by a method other than EZAZSSI or EZAZSSI timed out because it took too long to start TNF.

System action
TNF ends.

Operator response
Use EZAZSSI to start TNF.

System programmer response
None.

Module
MVPTNF

EZY6021I  TNF Stop Rejected - Active Users Exist

Explanation
Active users still exist for TNF; the stop is ignored.

System action
TNF continues.

Operator response
Either end the user address spaces or use the F  TNF , REMOVE... command to remove the users from the table.

System programmer response
None.

Module
MVPTNF

EZY6022I  TNF Stop Accepted

Explanation
The TNF STOP command was accepted.

System action
TNF stops.

Operator response
None.

System programmer response
None.
**EZY6023I**  TNF Modify Display Accepted

**Explanation**
The Modify Display command for TNF was accepted.

**System action**
TNF continues.

**Operator response**
None.

**System programmer response**
None.

---

**EZY6024I**  TNF Modify Remove Accepted

**Explanation**
The Modify Remove command for TNF was accepted.

**System action**
TNF continues.

**Operator response**
None.

**System programmer response**
This is probably an installation error. This message can be issued if resource manager load module MVPTTRML is not found in LPA. Ensure that the TCP/IP LPA load library SEZALPA is defined in one of the LPALSTxx members of SYS1.PARMLIB.

---

**EZY6025I**  TNF Modify Command Not Recognized

**Explanation**
The Modify command for TNF was not understood.

**System action**
TNF continues.

**Operator response**
For information on the Modify command, see the z/OS Communications Server: IP Diagnosis Guide.
System programmer response
None.

Module
MVPTNF

EZY6026I   TNF Start Accepted

Explanation
The TNF Start command was accepted.

System action
TNF continues.

Operator response
None.

System programmer response
None.

Module
MVPTNF

EZY6027I   TNF Command Not Recognized

Explanation
The command entered for TNF was not understood.

System action
TNF continues.

Operator response
For information on TNF commands, see the z/OS Communications Server: IP Diagnosis Guide.

System programmer response
None.

Module
MVPTNF

EZY6028I   TNF REMOVE not valid

Explanation
The Modify remove command for TNF was not understood.

System action
TNF continues.
**Operator response**
For information on the Modify command, see the *z/OS Communications Server: IP Diagnosis Guide*.

**System programmer response**
None.

**Module**
MVPTNF

**EZY6029I**  TNF REMOVELLLLLL not valid

**Explanation**
The Modify remove command for TNF was not understood.

**System action**
TNF continues.

**Operator response**
For information on the Modify command, see the *z/OS Communications Server: IP Diagnosis Guide*.

**System programmer response**
None.

**Module**
MVPTNF

**EZY6030I**  TNF All Users Removed

**Explanation**
The Modify remove,name=* was executed successfully.

**System action**
TNF continues.

**Operator response**
TNF can be ended now.

**System programmer response**
None.

**Module**
MVPTNF

**EZY6031I**  TNF User *userid* Not Found

**Explanation**
The Modify remove/display,name=*userid* was not found in the table.
**System action**
TNF continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPTNF

**EZY6032I TNF User userid Removed**

**Explanation**
The Modify remove,name=userid was found and removed.

**System action**
TNF continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPTNF

**EZY6033I TNF DISPLAY not valid**

**Explanation**
The Modify display command for TNF was not understood.

**System action**
TNF continues.

**Operator response**
For information on the Modify command, see the z/OS Communications Server: IP Diagnosis Guide.

**System programmer response**
None.

**Module**
MVPTNF

**EZY6034I TNF DISPLAYXXXXXX not valid**
**Explanation**
The Modify display command for TNF was not understood.

**System action**
TNF continues.

**Operator response**
For information on the Modify command, see the z/OS Communications Server: IP Diagnosis Guide.

**System programmer response**
None.

**Module**
MVPTNF

---

**EZY6035I**  TNF User *userid* Not Found

**Explanation**
The Modify remove,name=*userid* for TNF was not found in the table.

**System action**
TNF continues.

**Operator response**
Do a modify display to see if the user still exists.

**System programmer response**
None.

**Module**
MVPTNF

---

**EZY6036I**  TNF Active ASIDs

**Explanation**
This message is issued in response to the Modify display command. It is displayed when users exist.

**System action**
TNF continues.

**Operator response**
None.

**System programmer response**
None.
EZY6037I    Name ASID

Explanation
This message is issued in response to the Modify command. It is displayed when users exist.

System action
TNF continues.

Operator response
None.

System programmer response
None.

EZY6038I    userid asid

Explanation
This message is issued in response to the Modify command. It is displayed when users exist.

System action
TNF continues.

Operator response
None.

System programmer response
None.

EZY6039I    End

Explanation
This message is issued in response to the Modify command. It is displayed when users exist.

System action
TNF continues.

Operator response
None.
System programmer response
None.

Module
MVPTNF

EZY6040I  No TNF users exist

Explanation
This message is issued in response to the Modify command. It is displayed when no users exist.

System action
TNF continues.

Operator response
None.

System programmer response
None.

Module
MVPTNF

EZY6041I  TNF Modify Accepted

Explanation
The Modify command for TNF was accepted.

System action
TNF continues.

Operator response
None.

System programmer response
None.

Module
MVPTNF

EZY6042E  TNF ended: MVPTTRML Resource Manager Not Established

Explanation
A problem exists in the RESMGR interface.

System action
TNF ends.
Operator response
None.

System programmer response
This is probably an installation error.

Module
MVPTSSI

**EZY6043I**  TNF Start Initiated

Explanation
A TNF address space start was issued.

System action
EZAZSSI continues; start of TNF expected.

Operator response
None.

System programmer response
None.

Module
MVPTSSI

**EZY6044E**  Refused to create TNF address space

Explanation
The address space start from IEEMB881 failed.

System action
IPL continues.

Operator response
None.

System programmer response
Check to see that the data sets containing the respective load modules are cataloged and available.

Module
MVPTSSI

**EZY6045E**  TNF Not Initialized, Processing Continues

Explanation
The TNF address space start-up timed out. See the messages that usually accompany this message for an indication of the actual problem.
**System action**
IPL continues.

**Operator response**
None.

**System programmer response**
Correct the reason TNF did not start, and use EZAZSSI to restart TNF/VMCF.

**Module**
MVPTSSI

**EZY6046I**  
TNF Initialization Complete

**Explanation**
TNF is active and available.

**System action**
TNF continues, IPL continues.

**Operator response**
TNF applications can be started.

**System programmer response**
None.

**Module**
MVPTSSI

**EZY6047E**  
Subsystem name in IEFSSNxx is not TNF

**Explanation**
There is not a TNF entry in IEFSSNxx.

**System action**
TNF ends.

**Operator response**
None.

**System programmer response**
Add TNF entry to IEFSSNxx and re-IPL.

**Module**
MVPTSSI

**EZY6048E**  
VMCF ended: Node ID not specified in IEFSSNxx VMCF
Explanation
A nodename must be specified in the IEFSSNxx member for VMCF when using the migration implementation of restartable VMCF.

System action
IPL continues.

Operator response
Start EZAZSSI with a nodename after IPL completes.

System programmer response
For future IPLs add a nodename to the IEFSSNxx member.

Module
MVPXSSI

EZY6049I  VMCF Start Initiated

Explanation
A VMCF address space start was issued.

System action
Start of VMCF expected.

Operator response
None.

System programmer response
None.

Module
MVPXSSI

EZY6050E  Refused to create VMCF address space

Explanation
The address space start from IEEMB881 failed.

System action
IPL continues.

Operator response
Start EZAZSSI when the reason for the failure is corrected.

System programmer response
Check to see that the data sets containing the respective load modules are cataloged and available.
Module
MVPXSSI

**EZY6051E VMCF Not Initialized, Processing Continues**

**Explanation**
The VMCF address space start-up timed out. See the messages that usually accompany this message for an indication of the actual problem.

**System action**
IPL continues.

**Operator response**
Determine why VMCF did not start and start EZAZSSI when the IPL completes.

**System programmer response**
None.

Module
MVPXSSI

**EZY6052I VMCF Initialization Complete**

**Explanation**
VMCF is active and available.

**System action**
VMCF continues.

**Operator response**
VMCF applications can be started.

**System programmer response**
None.

Module
MVPXSSI

**EZY6053E Error creating PC numbers, is LPA correct?**

**Explanation**
PC numbers could not be created.

**System action**
VMCF ends.

**Operator response**
None.
System programmer response
This is probably an installation error. Check the LPA data sets.

Module
MVPXSSI

EZY6054E  VMCF ended: Error in initialization

Explanation
An error was encountered in initialization. See the messages that usually accompany this message for an indication of the actual problem.

System action
VMCF ends.

Operator response
Dependent on accompanying messages.

System programmer response
Dependent on accompanying messages.

Module
MVPXSSI

EZY6055E  Subsystem name in IEFSSNxx is not VMCF

Explanation
There is not a VMCF entry in IEFSSNxx.

System action
IPL Continues.

Operator response
None.

System programmer response
Add VMCF entries to IEFSSNxx and re-IPL.

Module
MVPXSSI

EZY6056I  VMCF not started correctly

Explanation
TNF was started by a method other than EZAZSSI or EZAZSSI timed out because it took too long to start VMCF.

System action
VMCF ends.
<table>
<thead>
<tr>
<th>Operator response</th>
<th>Use EZAZSSI to start TNF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>System programmer response</td>
<td>None.</td>
</tr>
<tr>
<td><strong>Module</strong></td>
<td>MVPXVI</td>
</tr>
<tr>
<td><strong>EZY6057I VMCF Stop Accepted</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Explanation</strong></td>
<td>The VMCF command was accepted.</td>
</tr>
<tr>
<td><strong>System action</strong></td>
<td>VMCF stops.</td>
</tr>
<tr>
<td><strong>Operator response</strong></td>
<td>None.</td>
</tr>
<tr>
<td><strong>System programmer response</strong></td>
<td>None.</td>
</tr>
<tr>
<td><strong>Module</strong></td>
<td>MVPXVI</td>
</tr>
<tr>
<td><strong>EZY6058I VMCF Modify Accepted</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Explanation</strong></td>
<td>The Modify command for VMCF was accepted.</td>
</tr>
<tr>
<td><strong>System action</strong></td>
<td>VMCF continues.</td>
</tr>
<tr>
<td><strong>Operator response</strong></td>
<td>None.</td>
</tr>
<tr>
<td><strong>System programmer response</strong></td>
<td>None.</td>
</tr>
<tr>
<td><strong>Module</strong></td>
<td>MVPXVI</td>
</tr>
<tr>
<td><strong>EZY6059I VMCF Modify Display Accepted</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Explanation</strong></td>
<td>The Modify display command for VMCF was accepted.</td>
</tr>
</tbody>
</table>
EZY6060I  VMCF Modify Remove Accepted

Explanation
The Modify Remove command for VMCF was accepted.

System action
VMCF continues.

Operator response
None.

System programmer response
None.

Module
MVPXVI

EZY6061I  VMCF Start Accepted

Explanation
The VMCF Start command was accepted.

System action
VMCF continues.

Operator response
None.

System programmer response
None.

Module
MVPXVI

EZY6062I  VMCF Command Not Recognized
**Explanation**  
The command entered for VMCF was not understood.

**System action**  
VMCF continues.

**Operator response**  
For information on the Modify command, see the z/OS Communications Server: IP Diagnosis Guide.

**System programmer response**  
None.

**Module**  
MVPXVI

---

**EZY6063I  VMCF Modify Command Not Recognized**

**Explanation**  
The Modify command for VMCF was not understood.

**System action**  
VMCF continues.

**Operator response**  
For information on the Modify command, see the z/OS Communications Server: IP Diagnosis Guide.

**System programmer response**  
None.

**Module**  
MVPXVI

---

**EZY6064I  VMCF Stop Rejected - Active Users Exist**

**Explanation**  
Active users still exist for VMCF; the stop is ignored.

**System action**  
VMCF continues.

**Operator response**  
Either terminate the user address spaces or use the TNF, REMOVE... command to remove the users from the table.

**System programmer response**  
None.
Module MVPXVI

EZY6065I VMCF REMOVE not valid

Explanation
The Modify remove command was not understood.

System action
VMCF continues.

Operator response
For information on the Modify command, see the z/OS Communications Server: IP Diagnosis Guide.

System programmer response
None.

Module MVPXVI

EZY6066I VMCF REMOVEXXXXXX not valid

Explanation
The Modify remove command was not understood.

System action
VMCF continues.

Operator response
For information on the Modify command, see the z/OS Communications Server: IP Diagnosis Guide.

System programmer response
None.

Module MVPXVI

EZY6067I VMCF All Users Removed

Explanation
The Modify remove,name=* was executed successfully.

System action
VMCF continues.

Operator response
VMCF can be stopped now.
**System programmer response**
None.

**Module**
MVPXVI

**EZY6068I**  VMCF User *userid* Not Found

**Explanation**
The Modify remove, name=*userid* was not found in the table.

**System action**
VMCF continues.

**Operator response**
Do a modify display to see if the user still exists.

**System programmer response**
None.

**Module**
MVPXVI

**EZY6069I**  VMCF User *userid* Removed

**Explanation**
The Modify remove, name=*userid* was found and removed.

**System action**
VMCF continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPXVI

**EZY6070I**  VMCF DISPLAY Invalid

**Explanation**
The Modify display command was not understood.

**System action**
VMCF continues.
**Operator response**
For information on the Modify command, see the z/OS Communications Server: IP Diagnosis Guide.

**System programmer response**
None.

**Module**
MVPXVI

**EZY6071I** VMCF DISPLAYXXXXXX not valid

**Explanation**
The Modify display command was not understood.

**System action**
VMCF continues.

**Operator response**
For information on the Modify command, see the z/OS Communications Server: IP Diagnosis Guide.

**System programmer response**
None.

**Module**
MVPXVI

**EZY6072I** VMCF User user id Not Found

**Explanation**
The Modify remove/display,name=user id was not found in the table.

**System action**
VMCF continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPXVI

**EZY6073I** VMCF Active ASIDs

**Explanation**
This message is issued in response to the Modify command. It is displayed when users exist.
System action
VMCF continues.

Operator response
None.

System programmer response
None.

Module
MVPXVI

EZY6074I Name ASID

Explanation
This message is issued in response to the Modify command. It is displayed when users exist.

System action
VMCF continues.

Operator response
None.

System programmer response
None.

Module
MVPXVI

EZY6075I userid asid

Explanation
This message is issued in response to the Modify command. It is displayed when users exist.

System action
VMCF continues.

Operator response
None.

System programmer response
None.

Module
MVPXVI

EZY6076I End
**Explanation**
This message is issued in response to the Modify command. It is displayed when users exist.

**System action**
VMCF continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPXVI

---

**EZY6077I**  No VMCF users exist

**Explanation**
This message is issued in response to the Modify command. It is displayed when no users exist.

**System action**
VMCF continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
MVPXVI

---

**EZY6101I**  fsswtr_id FSS=fss_addr, FSA=fsa_addr, FSD=fsd_addr, FSJ=fsj_addr

**Explanation**
This message is issued in conjunction with message EZY6102I at FSS startup time. Listed for the FSS specified in fsswtr_id are the addresses of the FSS, FSA, FSD and FSJ modules.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
Module
EZAPPFS

**EZY6102I  fsswtr_id FSID=fsid, ASID=asid, SUB=subsystem_id, OUT=route_code**

**Explanation**
This message is issued in conjunction with message EZY6101I at FSS startup time. Listed for the FSS specified in fsswtr_id are the FSID, ASID, Subsystem ID, and Route Code values.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

Module
EZAPPFS

**EZY6110I  fsswtr_id OPTS: option_1,option_2,...**

**Explanation**
This message is issued at FSS startup time. Multiple occurrences of the message are issued to list all FSS startup options in effect.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

Module
EZAPPFS

**EZY6111I  fsswtr_id STATS: statistic_1,statistic_2...**

**Explanation**
This message is issued when the FSS is modified. Multiple occurrences of the message are issued to list some statistics that might be useful for problem analysis.

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

System programmer response
None.

Module
EZAPPFS

EZY6112I  fsswtr_id FSS ENTERING WAIT

Explanation
The FSS named in the message has entered a wait state.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFS

EZY6113I  fsswtr_id FSS POSTED BY FSA TASK

Explanation
The FSS named in the message has been posted by an FSA.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFS

EZY6114I  fsswtr_id FSS POSTED BY JES

Explanation
The FSS named in the message has been posted by JES.
System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFS

EZY6115I  fsswtr_id FSS POSTED BY JES, ORDID=fss_order_id

Explanation
The FSS named in the message has been posted by JES, with the ORDER identified in the message.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFS

EZY6116I  fsswtr_id FSS POSTED BY COMMAND

Explanation
The FSS named in the message has been posted by an MVS operator console command.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFS

EZY6117I  fsswtr_id START-FSS CONNECT FSIREQ SUCCESSFUL
**Explanation**
The FSS-CONNECT request was successful for the FSS named in the message.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFS

---

**EZY6118I**  
`fsswtr_id STOP-FSS DISCONNECT FSIREQ SUCCESSFUL`

**Explanation**
The FSS-DISCONNECT request was successful for the FSS named in the message.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFS

---

**EZY6119I**  
`fsswtr_id START-FSA ATTACH SUCCESSFUL, FSID=fss_id`

**Explanation**
An FSA was successfully attached to the FSS named in the message. The FSS ID is also displayed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
**Module**
EZAPPFS

**EZY6120I**  
**fsswtr_id STOP-FSA DETACH SUCCESSFUL, FSID=fss_id**

**Explanation**
An FSA was successfully detached from the FSS named in the message. The FSS ID is also displayed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFS

**EZY6121I**  
**fsswtr_id FSS-AMODE/JES-AMODE/RESOLVED-CB-RMODE=fss_amode/jes_amode/cb_rmode**

**Explanation**
The control block RMODE that the FSS and FSA tasks will use (for the FSS named in the message) has been determined from the early FSS AMODE option setting. The RMODE setting will be permanent.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFS

**EZY6122I**  
**fsswtr_id INIT FUNCTION DONE**

**Explanation**
The initialization functions have been successfully completed for the FSS named in the message.

**System action**
Processing continues.

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

**Module**
EZAPPFS

**EZY6131E** fsswtr_id START-FSS START TOKEN INVALID: mgcr_token_addr

**Explanation**
The MGCR token address is invalid. The FSS cannot be started.

**System action**
Processing ends.

**Operator response**
The FSS cannot be explicitly started by the operator or run as a batch job. The FSS is started by JES.

**System programmer response**
None.

**Module**
EZAPPFS

**EZY6132E** fsswtr_id START-FSS ERROR, NO START CIB FOUND

**Explanation**
No START-verb CIB was found from the CSCB. This is an internal configuration error.

**System action**
Processing ends.

**Operator response**
Notify Systems Programming.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFS

**EZY6133E** fsswtr_id START-FSS START CMD INVALID, SUBSYS=subsystem_name

**Explanation**
An error was encountered in the FSS START PARM. The subsystem name is invalid. This is an internal configuration error.

**System action**
Processing ends.
Operator response
Notify Systems Programming.

System programmer response
Contact IBM Support.

Module
EZAPPFS

EZY6134E  fsswtr_id START-FSS START CMD INVALID, FSID=fss_id

Explanation
An error was encountered in the FSS START PARM. The functional subsystem is invalid. This is an internal configuration error.

System action
Processing ends.

Operator response
Notify Systems Programming.

System programmer response
Contact IBM Support.

Module
EZAPPFS

EZY6135E  fsswtr_id START-FSS START CMD INVALID, ROUTCDE=route_code

Explanation
An error was encountered in the FSS START PARM. The FSS-level message route code is invalid. This is an internal configuration error.

System action
Processing ends.

Operator response
Notify Systems Programming.

System programmer response
Contact IBM Support.

Module
EZAPPFS

EZY6136E  fsswtr_id START-FSS PARMLIB DD OPEN FAILED

Explanation
The data set defined by the PARMLIB DD JCL statement is known to exist but could not be opened.
System action
Processing ends.

Operator response
Determine why the data set cannot be opened.

System programmer response
None.

Module
EZAPPFS

EZY6137E  fsswtr_id LOAD OF EZAPPFCM FAILED

Explanation
The NPF file-create module could not be loaded.

System action
Processing ends.

Operator response
None.

System programmer response
Determine why LOAD failed for EZAPPFCM.

Module
EZAPPFS

EZY6138E  fsswtr_id INIT CALL FAILED IN FCM, RC=return_code

Explanation
The INIT function failed within the NPF file-create module.

System action
Processing ends.

Operator response
None.

System programmer response
Contact IBM Support.

Module
EZAPPFS

EZY6139E  fsswtr_id START-FSS CONNECT FSIREQ FAILED, SSOBRETN=fsi_return_code
**Explanation**
The FSS_CONNECT FSIREQ call failed.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFS

**EZY6140E**  
`fsswtr_id` START-FSS CONNECT FSIREQ FAILED, R15=ssreq_return_code

**Explanation**
The FSS_CONNECT FSIREQ call failed.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFS

**EZY6141E**  
`fsswtr_id` STOP-FSS REFUSED, FSAS STILL UP

**Explanation**
The FSS cannot SHUTDOWN while active FSAs are still present. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.
Module
EZAPPFS
EZY6142E  \textit{fsswtr\_id} STOP-FSS DISCONNECT FSIREQ FAILED, SSOBRETN=\textit{fsi\_error\_return\_code}

\textbf{Explanation}
The FSS did not DISCONNECT successfully. The return code from FSIREQ is listed. This is an internal error.

\textbf{System action}
Processing ends.

\textbf{Operator response}
None.

\textbf{System programmer response}
Contact IBM Support.

Module
EZAPPFS
EZY6143E  \textit{fsswtr\_id} STOP-FSS DISCONNECT FSIREQ FAILED, R15=\textit{register\_15}

\textbf{Explanation}
The FSS did not DISCONNECT successfully. The return code from the Subsystem Interface is listed. This is an internal error.

\textbf{System action}
Processing ends.

\textbf{Operator response}
None.

\textbf{System programmer response}
Contact IBM Support.

Module
EZAPPFS
EZY6144E  \textit{fsswtr\_id} STOP-FSS ABNORMAL DISCONNECT REQUESTED

\textbf{Explanation}
The FSS did not DISCONNECT successfully. The return code from the Subsystem Interface is listed. This is an internal error.

\textbf{System action}
Processing ends.

\textbf{Operator response}
None.
**System programmer response**
Contact IBM Support.

**Module**
EZAPPFS

---

**EZY6145E fswtr_id START-FSA-SDTE GETMAIN FAILED**

**Explanation**
GETMAIN was called for the FSA Daughter Task Element control block. The call failed.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Review any accompanying MVS system messages to try to determine if the GETMAIN failure was due to a correctable error. If the error cannot be corrected, contact IBM Support.

---

**Module**
EZAPPFS

---

**EZY6146E fswtr_id START-FSA ECBLIST GETMAIN FAILED**

**Explanation**
GETMAIN was called for the ECB list data area.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Review any accompanying MVS system messages to try to determine if the GETMAIN failure was due to a correctable error. If the error cannot be corrected, contact IBM Support.

---

**Module**
EZAPPFS

---

**EZY6147E fswtr_id START-FSA EP IDENTITY FAILED, R15=identify_return_code**

**Explanation**
The entry point for the FSA could not be identified. This is an internal error.

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

System programmer response
Contact IBM Support.

Module
EZAPPFS

EZY6148E  **fsuwtr_id** START-FSA SUBTASK ATTACH FAILED, R15=attach_return_code

Explanation
The FSA ATTACH failed. This is an internal error.

System action
Processing ends.

Operator response
None.

System programmer response
Contact IBM Support.

Module
EZAPPFS

EZY6149E  **fsuwtr_id** START-FSA ORDER REFUSED - sdte_fsid UP

Explanation
The FSA-START ORDER was refused. The FSA is already started. This is an internal error.

System action
Processing ends.

Operator response
None.

System programmer response
Contact IBM Support.

Module
EZAPPFS

EZY6150E  **fsuwtr_id** STOP-FSA SUBTASK DETACH FAILED, R15=detach_return_code

Explanation
The FSA DETACH failed. This is an internal error.
**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFS

**EZY6151E  fsswtr_id STOP-FSA ORDER REFUSED - fsid NOT UP**

**Explanation**
The FSA STOP order was processed, but the FSID could not be found. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFS

**EZY6152E  fsswtr_id UNSUPPORTED FSS ORDER, ORDID=fss_order_id**

**Explanation**
The FSS order could not be recognized or is invalid. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFS

**EZY6160W  fsswtr_id STOP (P) COMMANDS ARE NOT SUPPORTED AT THIS TIME**
**Explanation**
The FSS cannot be stopped via the command input buffer (P fswtr_id) in the current product release.

**System action**
Processing continues.

**Operator response**
If you wish to terminate the FSS, you must drain or stop the printer via the appropriate JES command.

**System programmer response**
None.

**Module**
EZAPPFS

---

**EZY6161W  fswtr_id UNRECOGNIZED COMMAND RECEIVED. REFUSED**

**Explanation**
The command received from the console command input buffer is not recognized as a valid command.

**System action**
Processing continues.

**Operator response**
Re-enter the command.

**System programmer response**
None.

**Module**
EZAPPFS

---

**EZY6162W  fswtr_id INVALID FSS OPTION NEAR: invalid_data**

**Explanation**
The option input is invalid.

**System action**
Processing continues.

**Operator response**
Re-input the option.

**System programmer response**
None.
Module
EZAPPFS

**EZY6163W**  
*fsrwtr_id* JES ASSUMING AMODE=31, AMODE=24 NOW INVALID

**Explanation**
The AMODE affecting existing tasks has been set to "31", and AMODE=24 cannot be used.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

Module
EZAPPFS

**EZY6164W**  
*fsrwtr_id* FSA TASK TERMINATED, FSID=*fss_id*

**Explanation**
The FSS has received notification that the FSA with the FSID displayed in the message text has ended.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

Module
EZAPPFS

**EZY6165W**  
*fsrwtr_id* FSA PRE-CONNECT FAILURE RESPONSE FSI-SENT, FSID=*fss_id*

**Explanation**
The FSS has sent notification via FSI SEND that the START FSA ORDER has failed.

**System action**
Processing continues.

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

Module
EZAPPFS

EZY6166W  fsswtr_id UNABLE TO GET FSIP AREA ABOVE 16MEG

Explanation
The FSS was unable to GETMAIN the needed storage area above the 16MB line.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFS

EZY6175I  fsswtr_id printer_id FSA FSID=fsid, UCB=ucb_name, AMODE=addr_mode

Explanation
This message is issued at FSA startup time. Listed for the FSA specified in printer_id for the FSS specified in fsswtr_id are the FSID, UCB, and AMODE.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6176I  fsswtr_id printer_id GDS job_number dd_name job_name separator_indicator

Explanation
This message is issued when a new data set is received by the FSS's FSA from JES. Job number, DD name, and job name are displayed, along with a print data set separator page indicator defined as follows:

NOSEP
  No separator page is required.
START
   A start data set separator page is required.

CONT
   A continue data set separator page is required.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6177I  fsswtr_id printer_id RDS job_number dd_name smf_indicator

Explanation
This message is issued if there is a data set at the stacker when RELDS is about to be issued. An indicator is issued at the end of the message as follows:

INCOMPLETE
   RELDS is incomplete.

SMF=YES
   An SMF type 6 record was requested.

SMF=NO
   No SMF type 6 record was requested.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6178I  fsswtr_id printer_id EOF/EOG job_number dd_name record_count TRLR/NOTRLR

Explanation
This message is issued to indicate that the data set has been marked for end-of-file or end-of-group. JES job number, DD name, and record count are shown in the message. The message will show either 'EOF' or 'EOG' between printer_id and job_number. 'TRLR' will appear at the end of the message to acknowledge the last data set in a JES2 output group or a JES3 job.
**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6179I**  
`fsswtr_id printer_id SND job_number dd_name OOP-TRACKING`

**Explanation**
This message indicates that a tracking FSI SEND operation has been performed. Job number and DD name appear in the message.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6180I**  
`fsswtr_id printer_id FSA ENTERING WAIT`

**Explanation**
The FSA associated with `printer_id` is waiting for work.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6181I**  
`fsswtr_id printer_id FSA POSTED BY JES`
**Explanation**
The FSI Order/Post ECB of the FSA associated with \((\text{printer_id})\) has been posted by JES.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6182I**  
\(fsswtr_id\) \(printer_id\) FSA POSTED BY JES, ORID=\(fsa\_order\_id\)

**Explanation**
The FSI Order/Post ECB of the FSA associated with \((\text{printer_id})\) has been posted by JES with the ORDER ID displayed in the message.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6183I**  
\(fsswtr_id\) \(printer_id\) FSA POSTED BY NPRO TIMER POP

**Explanation**
The NPRO STIMERM ECB of the FSA associated with \((\text{printer_id})\) has been posted.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
Module
EZAPPFA

EZY6184I fswtr_id printer_id FSA POSTED BY FSS

Explanation
The FSA associated with (printer_id) has been posted by the FSS.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6185I fswtr_id printer_id FSA DIRECTED BY FSS TO TERMINATE

Explanation
The FSA associated with (printer_id) has been posted by the FSS and directed to terminate.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6186I fswtr_id printer_id START-FSA CONNECT FSIREQ SUCCESSFUL

Explanation
The FSA-CONNECT FSIREQ request succeeded.

System action
Processing continues.

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

**Module**
EZAPPFA

**EZY6187I**  
*fsswtr_id printer_id* STOP-FSA DISCONNECT FSIREQ SUCCESSFUL

**Explanation**
The FSA-DISCONNECT FSIREQ request succeeded.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6188I**  
*fsswtr_id printer_id* DEVICE IS STARTING

**Explanation**
The FSA device is starting and is available for work.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6189I**  
*fsswtr_id printer_id* DEVICE IS STOPPING

**Explanation**
The FSA device is stopping and is not available for work.

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

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6190I  fsswtr_id printer_id OPERATOR INTERVENTION ORDERED**

**Explanation**
The FSA device requires operator intervention. The FSA has received an asynchronous ORDER request, via an FSIREQ, requesting that the FSA flush its device buffer and prepare for something that probably involves operator setup.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6191I  fsswtr_id printer_id DEVICE SYNC ORDERED**

**Explanation**
The FSA has an asynchronous ORDER request via an FSIREQ requesting that the FSA synchronize its processing to the point of actual printing. That is, a JES command such as a device backspace or forward space has been issued and it should affect the data set at the OOP (Operator Orientation Point).

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

**EZY6192I  fsswtr_id printer_id DEVICE SET ORDERED**
**Explanation**
The FSA has an asynchronous ORDER request via an FSIREQ requesting that the FSA reset some parameters. An example of this would be a JES2 $T command to change NPRO.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

EZY6193I  
*fswwtr_id* *printer_id* OUT OF BUFFERS IN GETREC, TRYING AGAIN

**Explanation**
The GETREC FSIREQ could not get the required number of buffers.

**System action**
The request will be reissued.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPFA

EZY6194I  
*fswwtr_id* *printer_id* NOTIFY SENT TO USER=*user_list*

**Explanation**
The user IDs listed in *user_list* were found in the NOTIFY parameter on the OUTPUT JCL statement. Data set processing results will be sent to the users in the list.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
Module
EZAPPFA

EZY6195I  \textit{fsrwtr\_id} printer\_id DATA SET AT OOP RESCHEDULED

Explanation
The data set at the Operator Orientation Point has been rescheduled.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6196I  \textit{fsrwtr\_id} printer\_id DATA SET AT OOP CANCELED

Explanation
The data set at the Operator Orientation Point has been cancelled.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6197I  \textit{fsrwtr\_id} printer\_id NO DATA SET AT OOP FOR SYNCH/OPINT

Explanation
A SYNCH or OPINT command was processed but no data set exists at the Operator Orientation Point.

System action
Processing continues.

Operator response
None.
System programmer response
None.

Module
EZAPPFA

EZY6198I  fswtr_id printer_id DEVICE BACK-SPACED

Explanation
The data set at the Operator Orientation Point has been backspaced.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6199I  fswtr_id printer_id DEVICE FORWARD-SPACED

Explanation
The data set at the Operator Orientation Point has been forward-spaced.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6200I  fswtr_id printer_id DEVICE SYNCH, ORDSYR1-4=sync_flags

Explanation
This message is a general DEVICE SYNCH message. It displays the flags from the JES-to-FSA SYNCH ORDER.

System action
Processing continues.
Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6201E  aaa bbbbbbbb ABENDED, ccccccc CODE=dddd REASON=eeeeeeee

Explanation
This message is presented by the NPF JES Capture Point ESTAE exit. The message contains information from an ABEND that occurred as follows:

aaa  Identifies the component that ABENDED. The value is either FSS or FSA.

bbbbbbbb  If aaa is FSS, bbbbbbbbb is the FSS writer ID. If aaa is FSA, bbbbbbbbb is the printer ID.

ccccccc  indicates either a SYSTEM or a USER ABEND.

dddddd  contains the ABEND code.

eeeeeeee  contains the reason code.

System action
Processing ends.

Operator response
None.

System programmer response
Use the information in this message as an aid to problem analysis.

Module
EZAPPFE

EZY6202E  EZAPPFE FSS/FSA SVC DUMP COMPLETE

Explanation
The SVC DUMP is complete. This message is issued from the NPF JES Capture Point ESTAE exit in conjunction with IEA794I.

System action
Processing ends.

Operator response
None.
System programmer response
Use the information in this message as an aid to problem analysis.

Module
EZAPPFE

EZY6203E  FSS/FSA SVC DUMP FAILED, R15=aaaa, REASON=bbbb

Explanation
The SVC DUMP failed. The return code from SDUMP (from register 15) is output in field aaaa. The reason code is output in field bbbb.

System action
Processing ends.

Operator response
None.

System programmer response
Use the information in this message as an aid to problem analysis.

Module
EZAPPFE

EZY6204E  LRB FAILING INSTR WAS aaaaaaaaaaa

Explanation
This message displays the Last-RB failing instruction in field aaaaaaaaaaa.

System action
Processing ends.

Operator response
None.

System programmer response
Use the information in this message as an aid to problem analysis.

Module
EZAPPFE

EZY6205E  LRB PSW = aaaaaaaaa aaaaaaaaa ILC = b IC = cc

Explanation
This message displays the Last-RB PSW in field aaaaaaaaa aaaaaaaaa, the Instruction Length Code in b, and the Interrupt Code in field cc.

System action
Processing ends.
Operator response
None.

System programmer response
Use the information in this message as an aid to problem analysis.

Module
EZAPPFE

**EZY6206E LRB aaa = register register register register**

Explanation
This message displays four Last-RB registers starting with the register identified in field `aaa`. `aaa` is one of the following: R0, R4, R8, or RC.

System action
Processing ends.

Operator response
None.

System programmer response
Use the information in this message as an aid to problem analysis.

Module
EZAPPFE

**EZY6207R REPLY "P" WHEN EZAPPFE MAY PERCOLATE, OR "TOE" (TIME-OF-ERROR) OR "LRB" (LAST-RB) FOR ERROR INFO RE-DISPLAY**

Explanation
This message enables you to display diagnostic data before the FSA or FSS task abends. EZAPPFE provides a basic ESTAE snapshot for these tasks. The operator must respond with either of the following replies:

- **P**
  - Causes EZAPPFE to percolate, which ends the FSA or FSS task, and takes an SVCDUMP.

- **TOE**
  - Redisplays the instruction at the time of the error, PSW, and registers, which can be used for diagnostic purposes.

- **LRB**
  - Redisplays the LAST RB information from the FSA or FSS task, which can be used for diagnostic purposes.

System action
The FSA or FSS task will end and an SVCDUMP will be taken.

Operator response
Save the system log and the dump for problem determination.

System programmer response
Check system log and console for related error messages.
Module
EZAPPFE

Procedure name
SFSEPI

EZY6214E  TOE FAILING INSTR WAS aaaaaaaaaaaa

Explanation
This message displays the Time-Of-Error failing instruction in field aaaaaaaaaaaa.

System action
Processing ends.

Operator response
None.

System programmer response
Use the information in this message as an aid to problem analysis.

Module
EZAPPFE

EZY6215E  TOE PSW = aaaaaaaaa aaaaaaaaa ILC = b IC = cc

Explanation
This message displays the Time-Of-Error PSW in field aaaaaaaaa aaaaaaaaa, the Instruction Length Code in b, and the Interrupt Code in field cc.

System action
Processing ends.

Operator response
None.

System programmer response
Use the information in this message as an aid to problem analysis.

Module
EZAPPFE

EZY6216E  LRB aaa = register register register register

Explanation
This message displays four Last-RB registers starting with the register identified in field aaa. aaa is one of the following: R0, R4, R8, or RC.

System action
Processing ends.
Operator response
None.

System programmer response
Use the information in this message as an aid to problem analysis.

Module
EZAPPFE

**EZY6221E**  
`fsswtr_id printer_id`  
FSA-CONNECT FSIREQ FAILED, SSOBRETN=`fsi_return_code`

Explanation
The FSA-CONNECT FSIREQ was unsuccessful. This is an internal error.

System action
Processing ends.

Operator response
None.

System programmer response
Contact IBM Support.

Module
EZAPPFA

**EZY6222E**  
`fsswtr_id printer_id`  
FSA-CONNECT FSIREQ FAILED, R15=`register_15_contents`

Explanation
The FSA-CONNECT FSIREQ was unsuccessful. This is an internal error.

System action
Processing ends.

Operator response
None.

System programmer response
Contact IBM Support.

Module
EZAPPFA

**EZY6223E**  
`fsswtr_id printer_id`  
INVALID UCB ADDRESS PASSED FROM JES

Explanation
The FSA detected that an invalid UCB address was passed from JES. This is an internal error.
**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
 EZAPPFA

**EZY6224E fsswtr_id printer_id GETMAIN FOR SMF WORK AREA FAILED**

**Explanation**
The FSA tried to issue GETMAIN for its SMF work area. The GETMAIN failed.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Review any accompanying MVS system messages to try to determine if the GETMAIN failure was due to a correctable error. If the error cannot be corrected, contact IBM Support.

**Module**
 EZAPPFA

**EZY6225E fsswtr_id printer_id GETMAIN FOR PIPELINE WORK AREAS FAILED**

**Explanation**
The FSA tried to issue GETMAIN for its PIPELINE work area. The GETMAIN failed.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Review any accompanying MVS system messages to try to determine if the GETMAIN failure was due to a correctable error. If the error cannot be corrected, contact IBM Support.

**Module**
 EZAPPFA

**EZY6226E fsswtr_id printer_id GETMAIN FOR FSI CKPT WORK AREA FAILED**
Explanation
The FSA tried to issue GETMAIN for its FSI checkpoint work area. The GETMAIN failed.

System action
Processing ends.

Operator response
None.

System programmer response
Review any accompanying MVS system messages to try to determine if the GETMAIN failure was due to a correctable error. If the error cannot be corrected, contact IBM Support.

Module
EZAPPFA

EZY6227E  FSSWTR_ID PRINTER_ID GETMAIN FOR SJF WORK AREA FAILED

Explanation
The FSA tried to issue GETMAIN for its Scheduler JCL Facility work area. The GETMAIN failed.

System action
Processing continues.

Operator response
None.

System programmer response
Review any accompanying MVS system messages to try to determine if the GETMAIN failure was due to a correctable error. If the error cannot be corrected, contact IBM Support.

Module
EZAPPFA

EZY6228E  GETMAIN FOR DEVICE DRIVER SDCB FAILED

Explanation
The FSA tried to issue GETMAIN for its Device Driver's Control Block. The GETMAIN failed.

System action
The FSA ends.

Operator response
None.

System programmer response
Review any accompanying MVS system messages to try to determine if the GETMAIN failure was due to a correctable error. If the error cannot be corrected, contact IBM Support.
Module
EZAPPFA

**EZY6230E**

`fsswtr_id printer_id FSA-DISCONNECT FSIREQ SSOBRETN=fsireq_return_code`

**Explanation**
The FSA did not DISCONNECT successfully. The return code from FSIREQ is listed. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

---

Module
EZAPPFA

**EZY6231E**

`fsswtr_id printer_id FSA-DISCONNECT FSIREQ R15=fsi_return_code`

**Explanation**
The FSA did not DISCONNECT successfully. The return code from the Subsystem Interface is listed. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

---

Module
EZAPPFA

**EZY6232E**

`fsswtr_id printer_id ABNORMAL FSA DISCONNECT REQUESTED`

**Explanation**
The FSA cannot DISCONNECT successfully. The time frame for disconnection is invalid. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.
System programmer response
Contact IBM Support.

Module
EZAPPFA

**EZY6233E  fsswtr_id printer_id FSA DISCONNECT REFUSED, DEVICE STILL UP**

**Explanation**
The FSA cannot DISCONNECT successfully. The device is still active. This is an internal error. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

System programmer response
Contact IBM Support.

Module
EZAPPFA

**EZY6234E  fsswtr_id printer_id START DEVICE ORDER REFUSED, ALREADY UP**

**Explanation**
The FSA has received an asynchronous ORDER request, via an FSIREQ, requesting that the FSA start its device. The FSA is already started. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

System programmer response
Contact IBM Support.

Module
EZAPPFA

**EZY6235E  fsswtr_id printer_id STOP DEVICE ORDER REFUSED, DEVICE NOT UP**

**Explanation**
The FSA has received an asynchronous ORDER request, via an FSIREQ, requesting that the FSA stop its device. The FSA is not started. This is an internal error.
**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

**EZY6236E**
`fsswtr_id printer_id UNSUPPORTED FSA ORDER, ORDID=fsa_order_id`

**Explanation**
The FSA has received an ORDER that is unsupported. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

**EZY6237E**
`fsswtr_id printer_id GETDS FSIREQ FAILED, R15=fsireq_return_code`

**Explanation**
A GETDS FSIREQ call failed. The return code from the call is listed in the message. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

**EZY6238E**
`fsswtr_id printer_id GETREC FSIREQ FAILED, R15=fsireq_return_code`
**Explanation**
A GETREC FSIREQ call failed. The return code from the call is listed in the message. This is an internal error.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

---

**EZY6239E**

**fsswtr_id printer_id**

FREEREC FSIREQ FAILED, R15=fsireq_return_code

**Explanation**
A FREEREC FSIREQ call failed. The return code from the call is listed in the message. This is an internal error.

**System action**
Data set processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

---

**EZY6240E**

**fsswtr_id printer_id**

CKPT FSIREQ FAILED, R15=fsireq_return_code

**Explanation**
A CKPT FSIREQ call failed. The return code from the call is listed in the message. This is an internal error.

**System action**
Data set processing ends.

**Operator response**
None.

**System programmer response**
Contact IBM Support.
EZY6241E  fsswtr_id printer_id GETREC I/O ERROR (GLRIOE)

Explanation
An I/O error occurred during GETREC processing.

System action
The data set is held and processing continues.

Operator response
None.

System programmer response
Analyze accompanying MVS system messages to determine the cause of the error. Contact IBM Support if you are unable to correct the error.

EZY6242E  fsswtr_id printer_id INVALID FSI PARMLIST IN GETREC (GLRIPL)

Explanation
During a GETREC FSIREQ, an invalid FSI parmlist was encountered. This is an internal error.

System action
Data set Processing ends.

Operator response
None.

System programmer response
Contact IBM Support.

EZY6243E  fsswtr_id printer_id GETREC PROCESSING ERROR (GLRLGE)

Explanation
During a GETREC FSIREQ, an internal logic error was encountered.

System action
The data set is held, and processing continues.

Operator response
None.
System programmer response
Contact IBM Support.

Module
EZAPPFA

EZY6244E  fsswtr_id printer_id DEVICE DRIVER EXIT ID/RC = exit_vector_offset/return_code

Explanation
A nonzero return code was received from the device driver exit.

System action
Processing continues.

Operator response
None.

System programmer response
Contact IBM Support.

Module
EZAPPFA

EZY6251W  fsswtr_id printer_id FSA REQUESTED TERMINATION (FIT SEND)

Explanation
An FSI SEND was issued to JES to request self-initiated termination.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAPPFA

EZY6252W  fsswtr_id printer_id SJF ERROR, SJFREQ R15=sjfreq_return_code

Explanation
An SJFREQ call failed, with the return code listed in the message.

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

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

**EZY6253W** `fsswtr_id printer_id SJF RETRIEVE ERROR, SJFREEREAS=sjfreq_reason_code`

**Explanation**
An SJF RETRIEVE error was detected that is not critical.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
On return from the SJF request, it was determined that some keywords were not found. This is assumed to be because the system JDTs are at a lower level than the list of keywords in the FSA module. Because that list is ordered by release date of keywords, it can be assumed that the needed information has been found.

**Module**
EZAPPFA

**EZY6255W** `fsswtr_id printer_id RELDS FSIREQ ERROR, RC=relds_return_code`

**Explanation**
A RELDS FSIREQ call was issued for the data set at the stacker. The call was unsuccessful.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

**EZY6256W** `fsswtr_id printer_id TRACKING FSI SEND ERROR, RC=return_code`

**Explanation**
A tracking SEND FSIREQ call was issued for the data set at the Operator Orientation Point. The call failed.
**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

**EZY6257W**  
_fswtr_id printer_id FSI SEND FSIREQ ERROR, RC=fsi_req_return_code_

**Explanation**
An FSISEND call was issued to JES, but was unsuccessful.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

**EZY6258W**  
_fswtr_id printer_id SJF UNKNOWN ERROR, SJFREREAS=sjfreq_reason_code_

**Explanation**
An SJFREQ call was unsuccessful. The reason could not be determined.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Contact IBM Support.

**Module**
EZAPPFA

**EZY6259W**  
_fswtr_id printer_id NOTIFY SSI ERROR, R15=ssi_return_code, USER=user_list_

572  z/OS Communications Server: IP Messages: Volume 3 (EZY)
Explanation
An attempt was made to notify a list of users that data set processing has been completed. The attempt was unsuccessful.

System action
Processing continues.

Operator response
None.

System programmer response
If you are unable to correct the conditions responsible for the NOTIFY failure, contact IBM Support.

Module
EZAPPFA

EZY6260W  fswtr_id printer_id NOTIFY ERROR, SSOBRETN=ssob_return_code

Explanation
An attempt was made to notify a list of users that data set processing has been completed. The attempt was unsuccessful.

System action
Processing continues.

Operator response
None.

System programmer response
If you are unable to correct the conditions responsible for the NOTIFY failure, contact IBM Support.

Module
EZAPPFA

EZY6261W  fswtr_id printer_id NOTIFY ERROR, SSNUERCD=ssnu_error_code

Explanation
An attempt was made to notify a list of users that data set processing has been completed. The attempt was unsuccessful.

System action
Processing continues.

Operator response
None.
System programmer response
If you are unable to correct the conditions responsible for the NOTIFY failure, contact the IBM Software Support Center to report this error.

Module
EZAPPFA
Chapter 9. EZYFxxx messages

EZYFS01I  SECURE_CTRLCONN value must be CLEAR, SAFE, or PRIVATE

Explanation
While processing the FTP.DATA file, the server or client encountered the SECURE_CTRLCONN statement with a parameter value that was not CLEAR, SAFE, or PRIVATE. The only valid values are CLEAR, SAFE, and PRIVATE.

System action
The line containing the error is ignored. FTP.DATA file processing continues with the next line of the file.

Operator response
Contact the system programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the statement. See the z/OS Communications Server: IP Configuration Guide for information about configuring the FTP server to use the TLS security mechanism.

Module
EZAFTPEP

Procedure name
read_ftpdata

EZYFS02I  SECURE_DATACONN value must be CLEAR, SAFE, PRIVATE, or NEVER

Explanation
While processing the FTP.DATA file, the server or client encountered the SECURE_DATACONN statement with a parameter value that was not CLEAR, SAFE, PRIVATE, or NEVER. The only valid values are CLEAR, SAFE, PRIVATE, and NEVER.

System action
The line containing the error is ignored. FTP.DATA file processing continues with the next line of the file.

Operator response
Contact the system programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the statement. See the z/OS Communications Server: IP Configuration Guide for information about configuring the FTP server to use the TLS security mechanism.

Module
EZAFTPEP

Procedure name
read_ftpdata
EZYFS03I  SECURE_MECHANISM value must be GSSAPI, or TLS

Explanation
While processing the FTP.DATA file, the client encountered the SECURE_MECHANISM statement with a parameter value that was not GSSAPI or TLS. The only valid values are GSSAPI or TLS.

System action
The line containing the error is ignored. FTP.DATA file processing continues with the next line of the file.

Operator response
Contact the system programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the statement. See the z/OS Communications Server: IP Configuration Guide for information about configuring the FTP server to use the TLS security mechanism.

Module
EZAFTPEP

Procedure name
read_ftpdata

EZYFS04I  Maximum allowable number of ciphersuite statements has been specified

Explanation
While processing the FTP.DATA file, the server or client encountered the CIPHERSUITE statement after the maximum number of valid CIPHERSUITE statements have been processed. Only 20 CIPHERSUITE statements can be processed.

System action
The line containing the statement is ignored. Processing of the FTP.DATA file continues with the next line of the file.

Operator response
Contact the System programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Delete the extra ciphersuite statements.

Module
EZAFTPEP

Procedure name
read_ftpdata

EZYFS05I  Ciphersuite name is not supported
**Explanation**
While processing the FTP.DATA file, the server or client encountered the CIPHERSUITE statement with a parameter value that was not recognized.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line of the file.

**Operator response**
Contact the System programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to contain the correct value for the statement. See the z/OS Communications Server: IP Configuration Guide for information about configuring the FTP server to use the TLS security mechanism.

**Module**
EZAFTPEP

**Procedure name**
read_ftpdata

**EZYFS06I  Ciphersuite name has already been specified**

**Explanation**
While processing the FTP.DATA file, the server or client encountered the CIPHERSUITE statement with a parameter name that has already been specified.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line of the file.

**Operator response**
Contact the System programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to contain no duplicate ciphersuite names.

**Module**
EZAFTPEP

**Procedure name**
read_ftpdata

**EZYFS07I  smfkeyword value must be TYPE119 or a blank**

**Explanation**
While processing the FTP.DATA file, the server encountered the SMF keyword parameter with a parameter value that was not TYPE119 or a blank. The only valid values for the smfkeyword parameter are TYPE119 and a blank. smfkeyword is the FTP.DATA keyword for SMF record.
**System action**
The line containing the SMF parameter is ignored. Processing continues with the next line in the file.

**Operator response**
Contact the system programmer with the error message.

**System programmer response**
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for more information about the FTP.DATA file.

**Module**
EZAFTPEP

**Procedure name**
read_ftpdata()

---

**EZYFS08I**  
**Some characters cannot be translated between codeset_1 and codeset_2**

**Explanation**
The SBDATACONN statement specified the code sets codeset_1 and codeset_2. The iconv() function was used to build a table for each of the 256 single byte character codepoints. Some of the codepoints do not have an equivalent codepoint.

**System action**
The tables are built and will be used during data transfer by FTP. If FTP detects a data byte during the transfer of the data that cannot be translated using the tables, FTP will use substitution characters if substitution is enabled, or otherwise fail the transfer.

**Operator response**
None.

**System programmer response**
Specify SBDATACONN with code set names that convert all of the codepoints. However, you can use the table if your data does not contain any of the untranslatable characters. To see which of the codepoints cannot be translated, start the client or server with the following trace specified in the FTP.DATA file:

```
DEBUG UTL ; utility services trace
```

**Module**
EZAFTPNX, EZAFTPCK

**Procedure name**
setup_translate_tables, locsite

---

**EZYFS09I**  
**SBSUBCHAR must be either SPACE or a hexadecimal character.**
**Explanation**
While processing the FTP.DATA file, the server encountered the SBSUBCHAR statement with a parameter value that was not SPACE, or a hexadecimal character. The only valid values for the SBSUBCHAR parameter are SPACE or a hexadecimal character.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

**Operator response**
Contact the System programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

**Procedure name**
read_ftpdata

---

**EZYFS10I**  
ENCODING value must be either SBCS or MBCS

**Explanation**
The ENCODING statement in the FTP.DATA file has a value that is not SBCS or MBCS. The only valid values are SBCS and MBCS.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

**Operator response**
Contact the system programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to contain the correct value for the specified statement. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

**Procedure name**
read_ftpdata

---

**EZYFS11I**  
Multi-byte encoding does not support codepage as a file system codepage

**Explanation**
The format of the MBDATACONN statement in the FTP.DATA file is the following:

```
MBDATACONN       (file_system_cp,network_transfer_cp).
```
The multi-byte encoding support for FTP allows codepages IBM-1388 and UTF-8 to be specified as the file system code page.

`codepage` is the code page name that was entered as the file system code page.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

**Operator response**
Contact the system programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to use one of the supported file system codepages. See the z/OS Communications Server: IP Configuration Reference for information about the MBDATACONN statement.

---

**EZYFS12I**  Multi-byte encoding does not support `codepage` as a network transfer codepage

**Explanation**
The format of the MBDATACONN statement in the FTP.DATA file is the following:

```
MBDATACONN (file_system_cp, network_transfer_cp).
```

The multi-byte encoding support for FTP allows codepage IBM-5488 to be specified as the network transfer code page.

`codepage` is the code page name that was entered as the network transfer code page.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

**Operator response**
Contact the system programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to use one of the supported network transfer codepages. See the z/OS Communications Server: IP Configuration Reference for information about the MBDATACONN statement.

---

**Module**
EZAFTPEP

**Procedure name**
verifyMBdataconn

---

**EZYFS13I**  No conversion available to `cp_name1` from `cp_name2`
Explanation
The MBDATACONN statement in the FTP.DATA file specified code page names, but there is no supported code set converter for the code sets (codepages) that are specified.

cp_name1 is the code page name to which the code is converted.
cp_name2 is the code page name from which the code is converted.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

Operator response
Contact the system programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to use one of the supported pairs of code sets (codepages). See the z/OS XL C/C++ Programming Guide for information about supported code set converters and valid code set names.

Module
EZAFTPEJ

Procedure name
verifyMBdataconn

EZYFS16I   SECURE_PASSWORD reset to REQUIRED

Explanation
After processing the FTP.DATA statements, the server cross-checked the values of the SECURE_LOGIN and SECURE_PASSWORD statements and found them to be in a combination that is not valid. SECURE_PASSWORD is coded with a value of OPTIONAL and SECURE_LOGIN has the value NO_CLIENT_AUTH. NO_CLIENT_AUTH indicates that the server does not request a certificate from the client, but SECURE_PASSWORD OPTIONAL means that a certificate is required for a session protected by the TLS security mechanism.

System action
Because the server will not request a certificate from the client, SECURE_PASSWORD is reset to REQUIRED. As a result, a password will be required for authentication of a TLS secured login.

Operator response
If the new value is acceptable, no action is required. Otherwise, contact the system programmer to change the SECURE_LOGIN setting to a value other than NO_CLIENT_AUTH in order to permit SECURE_PASSWORD OPTIONAL.

System programmer response
If necessary, update the FTP.DATA file with the correct values for SECURE_LOGIN and SECURE_PASSWORD. See z/OS Communications Server: IP Configuration Reference for information about the statements of the FTP.DATA file.

Module
EZAFTPEP
**Procedure name**
read_ftpdatal

**EZYFS201**  Binary tagged file translated with current data connection translation table

**Explanation**
The file that is to be transferred was tagged binary but the data type is ASCII. This is a warning to the user that the file will be translated.

**System action**
The file transfer continues.

**Operator response**
If it is acceptable that the translation was done in ASCII, then no action is necessary. However, if no translation should have been done, change the transfer type to binary and transfer the file again. See the z/OS UNIX System Services Command Reference for information about the CHTAG command. See the z/OS Communications Server: IP User’s Guide and Commands for information about the FTP TYPE subcommand.

**System programmer response**
None

**Module**
EZAFTPSM

**Procedure name**
hfs_sndFile()

**EZYFS241**  SECURE_HOSTNAME value must be OPTIONAL or REQUIRED

**Explanation**
While processing the FTP.DATA file, the client encountered the SECURE_HOSTNAME statement with a parameter value that was not either REQUIRED or OPTIONAL. The only valid values are REQUIRED or OPTIONAL.

**System action**
The line containing the error is ignored. FTP.DATA file processing continues with the next line of the file.

**Operator response**
Contact the system programmer to correct the SECURE_HOSTNAME statement in the FTP.DATA file.

**System programmer response**
Correct the SECURE_HOSTNAME statement in the FTP.DATA file.
See the SECURE_HOSTNAME statement in z/OS Communications Server: IP Configuration Reference for more information.

**Module**
EZAFTPEP
Procedure name
read_ftpdata

EZYFS30W FTP message catalog catligname returned an unexpected timestamp of timestamp1 - FTP expected timestamp2 - FTP will use default messages

Explanation
The time stamp that is contained in the catalog does not match the time stamp that FTP expects.

In the message text:

*catligname*
The fully qualified name of the catalog that FTP is processing.

*timestamp1*
The time stamp from the catalog, specified in the following format:

```
yyyy ddd hh:mm UTC
```

- *yyyy* is the year
- *ddd* is the day (001 - 366)
- *hh* is the hour (01 - 24)
- *mm* is the minute (01 - 60)

*timestamp2*
The time stamp that FTP expects to be in the catalog. This time stamp in the same format as the *timestamp1* value.

System action
FTP uses the default messages. Message EZYFS32I follows this message and indicates the expected service level.

Operator response
No action needed.

System programmer response
Ensure that the correct z/OS UNIX file system is attached and that the correct service level is applied to the z/OS UNIX file system. If a customized catalog is being used, verify that it is at the same service level as the current distribution level of the catalog that is specified in message EZYFS32I.

User response
Contact the system programmer.

Problem determination
See the system programmer response.

Source
z/OS Communications Server TCP/IP: FTP

Module
EZAFTPCY
Routing code
10

Descriptor code
3

Example
EZYFS30W FTP message catalog /usr/lib/nls/msg/C/ftpdmsg.cat returned an unexpected timestamp of 2005.180 15:30 UTC - FTP expected 2006.091 06:30 UTC - FTP will use default messages

EZYFS31W FTP reply catalog catligname returned an unexpected timestamp of timestamp1 - FTP expected timestamp2, - FTP will use default messages

Explanation
The time stamp that is contained in the catalog does not match the time stamp that the FTP application expects. In the message text:
catligname
The name of the catalog that FTP opened.
timestamp1
The timestamp from the catalog, specified in the following format:

 yyyy ddd hh:mm UTC

• yyyy is the year
• ddd is the day (001-366)
• hh is the hour (01-24)
• mm is the minute (01-60)
timestamp2
The time stamp that FTP expects to be in the catalog. This time stamp is in the same format as the timestamp1 value.

System action
FTP uses the default messages. Message EZYFS32I follows this message and indicates the expected service level.

Operator response
No action needed.

System programmer response
Ensure that the correct z/OS UNIX file system is attached and that the correct service level is applied to the z/OS UNIX file system. If a customized catalog is being used, verify that it is at the same service level as the current distribution level of the catalog that is specified in EZYFS32I.

User response
Contact the system programmer

Problem determination
See the system programmer response.
**Example**

```
EZYFS31W FTP reply catalog /usr/lib/nls/msg/C /ftpdrply.cat returned an unexpected timestamp of 2005 180 06:35 UTC - FTP expected 2006 091 06:45 UTC - FTP will use default messages
```

**EZYFS32I** The catalog *catlgname* must be at service level *svclevel*

**Explanation**

The specified message catalog is not at the expected service level. Message EZYFS30W or message EZYFS31W precedes this message and identifies the time stamps that resulted in the detection of a catalog mismatch.

In the message text:

- **catlgname** The catalog name.
- **svclevel** The service level (FMID or PTF) expected by the catalog.

**System action**

FTP uses the default messages.

**Operator response**

No action needed.

**System programmer response**

Ensure that one of the following is true:

- The correct z/OS UNIX file system is attached
- The correct service has been applied to the z/OS UNIX file system
- A customized catalog is being used and is at the same level as the distribution level of the catalog that is specified in the message.

**User response**

Contact the system programmer.

**Problem determination**

See the system programmer response.
Example

In the following example, the value UK91456 specifies the catalog that shipped as part of PTF UK91456.

EZYFS32I The catalog /usr/lib/nls/msg/C /ftpdrply.msg must be at service level UK91456

In the following example, the value HIP6190 specifies that the catalog should be at the same release level.

EZYFS32I The catalog /usr/lib/nls/msg/C /ftpdmsg.msg must be at service level HIP6190

EZYFS33I FTP will remove type sequence numbers from input commands

Explanation

FTP is configured with the value TRUE on the SEQNUMSUPPORT parameter in the FTP.DATA file. Sequence numbers that have numerics in columns 1 - 8 on the input command (LEADING) are shifted left by 8 columns prior to processing the command. Sequence numbers that have numerics in the last 8 columns on the input command (TRAILING) have the last 8 columns replaced with blanks.

In the message text:

**type**

The type of sequence numbers detected. Possible values are:

**LEADING**

Indicates that columns 1 - 8 of the input contain numeric data; FTP treats this and subsequent records as having a sequence number in these columns. The following is an example of leading sequence numbers:

| Input command: 00000130 cd /tmp | Processed command: cd /tmp |

**TRAILING**

Indicates that the last 8 columns of data that was read contains numeric data; FTP treats this and subsequent records as having a sequence number in these columns. The following is an example of trailing sequence numbers:

| Input command: cd /tmp 00000130 | Processed command: cd /tmp |

System action

FTP continues.

Operator response

No action needed.
**System programmer response**

This message is issued only when the SEQNUMSUPPORT TRUE value is in effect. Ensure that the SEQNUMSUPPORT TRUE statement in the FTP.DATA file is appropriate for your installation.

**User response**

Do nothing if you intended to remove sequence numbers and FTP processes the input without errors. If you must deactivate FTP sequence number support, perform one of the following actions:

- If you are making a permanent change, contact the system programmer.
- When FTP runs in batch mode and you cannot change the FTP.DATA file, temporarily concatenate the required command to the FTP.DATA file:

```
//SYSFTPD DD DSN=SYS1.TCPPARMS(FTPDATA),DISP=SHR CURRENT FTP.DATA FILE
// DD *
SEQNUMSUPPORT FALSE
/*
```

**Problem determination**

See the system programmer response.

**Source**

z/OS Communications Server TCP/IP: FTP

**Module**

EZYFTPCU

**Routing code**

10

**Descriptor code**

12

**Example**

EZYFS33I FTP will remove LEADING sequence numbers from input commands
EZYFS33I FTP will remove TRAILING sequence numbers from input commands

**EZYFS34W FTP will not remove type sequence numbers**

**Explanation**

FTP is configured with the value FALSE on the SEQNUMSUPPORT parameter in the FTP.DATA file but FTP has detected possible sequence numbers in the input. These sequence numbers are not removed; however, this situation can cause an error in FTP processing.

In the message text:

**type**

The type of sequence numbers that were detected. Possible values are:

- **LEADING**
  
  A sequence number was detected in columns 1 - 8 of the input record.

- **TRAILING**
  
  A sequence number was detected in the last 8 bytes of the input record.
System action
FTP continues processing the command.

Operator response
No action needed.

System programmer response
To ignore sequence numbers, add the SEQNUMSUPPRT TRUE parameter to the FTP.DATA file.

User response
Determine whether the input FTP subcommands contain sequence numbers. If the input does contain sequence numbers, do one of the following:

• Use an editor and remove sequence numbers.
• Add the SEQNUMSUPPRT TRUE value to the FTP.DATA file.
• When FTP runs in batch mode and you cannot change the FTP.DATA file, temporarily concatenate the required command to the FTP.DATA file:

```
//SYSFTPD DD DSN=SYS1.TCPPARMS(FTPDATA),DISP=SHR CURRENT FTP.DATA FILE
//        DD *
SEQNUMSUPPRT TRUE
/*
```

Problem determination
None.

Source
z/OS Communications Server TCP/IP: FTP

Module
EZAFTPCU

Routing code
10

Descriptor code
12

Example
EZYFS34W FTP will not remove LEADING sequence numbers
EZYFS34W FTP will not remove TRAILING sequence numbers

EZYFS35I FTP will not remove sequence numbers from input

Explanation
FTP is configured with the value TRUE on the SEQNUMSUPPRT parameter in the FTP.DATA file. While processing an input file, FTP detected a record that indicated that FTP should stop removing sequence numbers. See the SEQNUMSUPPRT statement in z/OS Communications Server: IP Configuration Reference.
System action
FTP processing continues.

Operator response
No action needed.

System programmer response
None.

User response
If a command fails after this message was issued, ensure that the command does not have a sequence number that was interpreted as part of the command.

Problem determination
None.

Source
z/OS Communications Server TCP/IP: FTP

Module
EZAFTPCU

Routing code
10

Descriptor code
12

Example
None.

EZYFS50I  ID=sessionID  CONN starts  Client IPaddr=ipaddr  hostname=hostname

Explanation
This log entry is made by the FTP daemon when it accepts a client connection request. The keyword CONN identifies this entry as a connection log entry.

sessionID uniquely identifies the FTP session between a client and a server. The identifier is created by combining the jobname of the FTP daemon with a 5-digit number in the range 00000–99999. This identifier is in each log entry for the session until message EZYFS52I, which is the last entry for the session.

ipaddr is the IP address of the FTP client. The IP address might be either an IPv4 or an IPv6 address.

hostname is the name of the FTP client. If the name cannot be resolved, UNKNOWN is displayed.

System action
FTP continues.

Operator response
None.
System programmer response
None.

Module
EZAFTPBU

Procedure name
logCONN

EZYFS51I  ID=sessionID  CONN fails  Reason=reason  Text=text

Explanation
This log entry is made by the FTP daemon or the server when a connection request fails.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

reason is a number code for the failure.

text is an explanation of the failure.

System action
The FTP connection ends.

Operator response
None.

System programmer response
Use the reason code in the log entry to choose from the following responses:

1
   Examine your FTCHKIP user exit to determine why the connection was rejected.

2
   If the FTP server trace was not active with the INT option, use the MODIFY operator command to activate the trace with the INT option. Ask the client to connect again to the server. Contact the IBM support center and provide the log entry message number, the reason number of the failure, and the FTP server trace of the failure.

3
   If the FTP server trace was not active with the INT option, use the MODIFY operator command to activate the trace with the INT option. Ask the client to connect again to the server. Contact the IBM support center and provide the log entry message number, the reason number of the failure, and the FTP server trace of the failure.

4
   If the FTP server trace was not active with the INT option, use the MODIFY operator command to activate the trace with the INT option. Ask the client to connect again to the server. Contact the IBM support center and provide the log entry message number, the reason number of the failure, and the FTP server trace of the failure.

5
   If the FTP server trace was not active with the INT option, use the MODIFY operator command to activate the trace with the INT option. Ask the client to connect again to the server. Contact the IBM support center and provide the log entry message number, the reason number of the failure, and the FTP server trace of the failure.
If the FTP server trace was not active with the INT option, use the MODIFY operator command to activate the trace with the INT option. Ask the client to connect again to the server. Contact the IBM support center and provide the log entry message number, the reason number of the failure, and the FTP server trace of the failure.

Restart the daemon with a larger region size and ask the client to connect again to the server.

Restart the daemon with a larger region size and ask the client to connect again to the server.

Restart the daemon with a larger region size and ask the client to connect again to the server.

Contact the IBM support center and provide the log entry message number and reason number of the failure.

Restart the daemon with a larger region size and ask the client to reconnect to the server.

FTP sessions that are created as a result of a port scanner application testing for server response are likely to have this message logged for those sessions. The reason code will vary depending on the timing of the scanner's disconnection process, with codes 2-6 being the most likely. To verify that a test for server response is the cause, check for that application running on the IP address reported in the associated EZYFS50I message and verify that the times correlate with its activities.

Module
EZAFTPBU

Procedure name
logCONN

EZYFS52I  ID=sessionID  CONN ends  Input=bytesIn  Output=bytesOut

Explanation
This log entry is made by the FTP server when an FTP session with the client ends. The keyword CONN identifies this entry as a connection log entry.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

bytesIn is the count, in bytes, of the data that was transferred into the server with the data transfer commands (STOR, STOU, and APPE).

bytesOut is the count, in bytes, of the data that was transferred from the server with the data transfer command (RETR) and the list commands (LIST and NLST).

If a count is greater than or equal to one gigabyte (that is, 1,073,741,824 bytes), the count is displayed as a number of gigabytes rounded to the nearest hundredth of a gigabyte. Some examples are:

- 1,073,741,824 bytes is displayed as 1.00 GB
- 2,147,483,648 bytes is displayed as 2.00 GB
- 2,415,919,104 bytes is displayed as 2.25 GB

System action
The FTP connection ends.

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

**Module**
EZAFTPBU

**Procedure name**
logCONN

**EZYFS54I**  
**ID=**sessionID **SECURE OK Mechanism=**mechanism

**Explanation**
This log entry is made by the FTP server to indicate that the connection with the client is protected by a security mechanism.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

mechanism is either TLS, TLS-P, or GSSAPI.

**System action**
FTP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPBU

**Procedure name**
logSECURE

**EZYFS55I**  
**ID=**sessionID **SECURE fails Reason=**reason **Text=**text

**Explanation**
This log entry is made by the FTP server when a request for security protection fails.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

reason is a number code for the failure

text is an explanation of the failure.

**System action**
FTP continues.

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

Use the reason code number in the message to determine your action as follows:

1  If your installation wants to provide TLS security for FTP connections, add the EXTENSIONS AUTH_TLS and the KEYRING statements to your FTP.DATA file and restart the FTP daemon. Otherwise, use the message EZYFS50I for this session ID to determine the identity of the FTP client that is requesting a secure session and handle appropriately for your installation.

2  If your installation wants to allow private data connections with TLS secure connections, change the SECURE_DATACONN statement in your FTP.DATA file to CLEAR or PRIVATE and restart the FTP daemon. Otherwise, use the message EZYFS50I for this session ID to determine the identity of the FTP client that is requesting a secure session with private data connections and handle appropriately for your installation.

3  If your installation wants to provide Kerberos security for FTP connections, add the EXTENSIONS AUTH_GSSAPI statement to your FTP.DATA file and restart the FTP daemon. See the z/OS Communications Server: IP Configuration Guide for more information. Otherwise, use the message EZYFS50I for this session ID to determine the identity of the FTP client that is requesting a secure session and handle appropriately for your installation.

4  Use the MODIFY operator command to set the FTP server trace as follows: MODIFY jobname,DEBUG=(SOC(3),SEC). Then ask the client to try to connect to the FTP server again. Contact the IBM support center and provide the server trace and the log entry message.

5  If your installation wants to allow the USER command before the session is protected by a security mechanism, add the SECURE_FTP ALLOWED statement to the FTP.DATA file and restart the daemon. If you require the security protection, then the client must send an AUTH command before the USER command is sent.

6  If your installation wants to allow the PASS command before the session is protected by a security mechanism, add the SECURE_FTP ALLOWED statement to the FTP.DATA file and restart the daemon. If you require the security protection, then the client must send an AUTH command before the PASS command is sent.

7  If your installation wants to allow commands before the session is protected by a security mechanism, add the SECURE_FTP ALLOWED statement to the FTP.DATA file and restart the daemon. If you require the security protection, then the client must send an AUTH command as the first command.

8  If GSSAPI connections are required, provide an FTP server that supports the IPv4 transport.

**Module**

EZAFTPBU

**Procedure name**

logSECURE

**EZYFS56I  ID=sessionID ACCESS OK USERID=userid**

**Explanation**

This log entry is made by the FTP server to indicate a successful verification of the user that logged in with the USER command. If the client changes the user ID during the session with another USER command, another ACCESS entry is made to record the new user ID.
sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

userid is the name that was entered on the USER command.

**System action**
FTP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPBU

**Procedure name**
logACCESS

**EZYFS57I**  ID=sessionID ACCESS fails USERID=userid Reason=reason_code Text=text

**Explanation**
This log entry is made by the FTP server when the access verification fails.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

userid is the login name (user ID).

reason_code is the code that corresponds to the failure.

text is an explanation of the failure.

**System action**
FTP continues. The client is still in session but is not logged in.

**Operator response**
None.

**System programmer response**
Use the reason_code, described in the following table, to determine your response. If the identity of the client is needed, use the log entry message EZYFS50I for this session ID to determine the identity of the FTP client that is accessing the FTP server with the USER or PASS command. If an FTP server trace is needed, ensure that the ACC trace option is activated for the server trace before the client logs in. Use the MODIFY operator command or an FTP.DATA file statement to activate the trace.

<table>
<thead>
<tr>
<th>reason_code</th>
<th>System programmer response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ask the client to reissue the USER and PASS commands, using the correct format of the <code>old_password/new_password/new_password</code> on the PASS command.</td>
</tr>
<tr>
<td>2</td>
<td>Ask the client to reissue the USER and PASS commands, using the same password for both occurrences of <code>new_password</code>.</td>
</tr>
<tr>
<td>reason_code</td>
<td>System programmer response</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>3</td>
<td>The FTCHKPWD user exit of your installation rejected the user ID. Update the exit to accept the user ID.</td>
</tr>
<tr>
<td>4</td>
<td>Ask the client to reissue the USER and PASS commands, using the correct password for the user ID.</td>
</tr>
<tr>
<td>5</td>
<td>Ask the client to reissue the USER and PASS commands with valid user ID and password values.</td>
</tr>
<tr>
<td>6</td>
<td>Ensure that programs being loaded from the address space of the FTP server are defined as program controlled. A corresponding ICH420I message is issued to identify the uncontrolled library where the load was done.</td>
</tr>
<tr>
<td>7</td>
<td>Ask the client to reissue the USER and PASS commands, using the format of <code>old_password/new_password/new_password</code> on the PASS command to provide a new password for the user ID.</td>
</tr>
<tr>
<td>8</td>
<td>Ask the client to reissue the USER and PASS commands, using the format of <code>old_password/new_password/new_password</code> on the PASS command with a new password that meets the installation’s requirements for passwords.</td>
</tr>
<tr>
<td>9</td>
<td>Typically, this code indicates that user access was denied by SAF services. For example, a user tried to log in to FTP with a user ID that was revoked. If your security product is IBM RACF, you can use the <code>reason_code</code> value associated with this code to determine why the user access was denied. The <code>reason_code</code> value contains the RACF return code and reason codes from the RACROUTE REQUEST=EXTRACT service. See the system macro information in z/OS Security Server RACROUTE Macro Reference for descriptions of the return and reason codes.</td>
</tr>
<tr>
<td>10</td>
<td>Determine why the user ID is no longer known, or ask the client to reissue the USER and PASS commands with a user ID that is known to the system.</td>
</tr>
<tr>
<td>11</td>
<td>Access to the user database by user ID is a function provided by the getpwnam() function of the operating system. Contact the IBM support center and provide the log entry message number, reason number of the failure, and the FTP server trace for the failure.</td>
</tr>
<tr>
<td>12</td>
<td>The process group ID is set by the setgid() function of the operating system. If the FTP server ACC trace was active when the setgid() failed, the server made a trace entry with the errno of the failure. The errno is 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. Contact the IBM support center and provide the log entry message number, reason number of the failure, and the FTP server trace for the failure.</td>
</tr>
<tr>
<td>13</td>
<td>Contact the IBM support center and provide the log entry message number, reason number of the failure, and the FTP server trace for the failure.</td>
</tr>
<tr>
<td>14</td>
<td>Contact the IBM support center and provide the log entry message number, reason number of the failure, and the FTP server trace for the failure.</td>
</tr>
<tr>
<td>15</td>
<td>Contact the IBM support center and provide the log entry message number, reason number of the failure, and the FTP server trace for the failure.</td>
</tr>
<tr>
<td>16</td>
<td>Contact the IBM support center and provide the log entry message number, reason number of the failure, and the FTP server trace for the failure.</td>
</tr>
<tr>
<td>reason_code</td>
<td>System programmer response</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------</td>
</tr>
</tbody>
</table>
| 17          | Your installation has defined in the security product (for example, RACF) a SERVAUTH class and a profile for the port as follows: EZB.FTP.systemname.ftpd daemonname.PORTxxxxx where xxxxx is the port number of the FTP daemon. The user login is denied because of the following:  
  - The session is protected by TLS.  
  - The FTP.DATA file has the statement SECURE_LOGIN VERIFY_USER.  
  - The user ID does not have read access to the profile.  
  Give the user ID read access to the profile if the user needs to use TLS security. |
| 18          | The connection is protected by TLS security. However, the client did not send a certificate to the FTP server during the TLS handshake process. Either the client must send a certificate or you should change the FTP.DATA file statement to SECURE_LOGIN NO_CLIENT_AUTH. |
| 19          | The USER command is disabled for this client. Inform the client that he must end his current FTP session and connect again to provide the USER command and the PASS command with the correct password. |
| 20          | To allow anonymous logins with an active security mechanism, code the ANONYMOUSLEVEL statement in FTP.DATA file with a value greater than 2. |
| 21          | Determine whether the certificate or ticket used to authenticate the user is associated in the security product with a different user. |
| 22          | The server received a PASS command from the client that did not include a password. If your FTP client is z/OS, enter a password when prompted by the FTP client. For other FTP clients, consult vendor documentation. |
| 23          | To allow clients to switch between users, code the ANONYMOUSLEVEL statement in the FTP.DATA file with a value less than 3. |
| 24          | To allow anonymous logins with the initial filetype, code the ANONYMOUSFILETYPEJES, ANONYMOUSFILETYPESEQ, and ANONYMOUSFILETYPESQL statements to be consistent with the FILETYPE statement in the FTP.DATA file. |
| 25          | To allow anonymous logins, code the ANONYMOUSFILEACCESS statement to be consistent with the STARTDIRECTORY statement in the FTP.DATA file. |
| 26          | Ask the client to reissue the USER and PASS commands by using the correct format of e-mail address on the PASS command. |
| 27          | Ask the client to reissue the USER command with a valid user name. |
| 28          | The connection is protected by Kerberos security. Determine whether the ticket used to authenticate the user is associated in the security product with a different user. |

**Module**  
EZAFTPBU

**Procedure name**  
logACCESS

**EZYFS58I**  
ID=sessionID ALLOC permission denied MVS DSN=dsname
Explanation
This log entry is made by the FTP server when it determines through the use of a SAF-compliant security product (for example, RACF) that the end user does not have permission to access the MVS data set for the command that is being processed.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

dsnname is the full MVS name of the data set.

System action
The MVS data set is not allocated.

Operator response
None.

System programmer response
If the user should be allowed to access the data set, update the security product to give permission for access to the MVS data set to the user identified in the log entry message EZYFS56I. Otherwise, use the message EZYFS50I for this session ID to determine the identity of the FTP client that is requesting access to the data set and handle appropriately for your installation.

Module
EZAFTPB

Procedure name
logALLOC

EZYFS59I ID=sessionID ALLOC permission denied HFS filename=filename

Explanation
This log entry is made by the FTP server when it determines that the end user does not have permission to access the z/OS UNIX file for the command that is being processed.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

filename is the full z/OS UNIX file name of the file.

System action
The z/OS UNIX file is not allocated.

Operator response
None.

System programmer response
If the user should be allowed to access the file, update the permissions of the file and any directories in the path to allow access to the z/OS UNIX file for the user identified in the log entry message EZYFS56I. Otherwise, use the message EZYFS50I for this session ID to determine the identity of the FTP client that is requesting access to the file and handle appropriately for your installation.
**Module**
EZAFTPBU

**Procedure name**
logALLOC

**EZYFS60I**  
ID=sessionID ALLOC OK  
*action* MVS DSN=dsname

**Explanation**
This log entry is made by the FTP server when it successfully allocates an MVS data set.

*sessionID* uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

*action* indicates whether a new data set was created during allocation or an existing data set was used. The values for action are the following:

**Create**
A new data set was created.

**Use**
An existing data set was used.

*dsname* is the full MVS name of the data set.

**System action**
FTP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPBU

**Procedure name**
logALLOC

**EZYFS61I**  
ID=sessionID ALLOC DDNAME=ddname VOLSER=volser DSORG=dsorg DISP=(disp)

**Explanation**
This log entry provides additional information about an MVS data set allocation.

*sessionID* uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

*ddname* is the ddname associated with the allocation.

*volser* is the first volume serial number associated with the allocation.

*dsorg* is the data set organization of the data set. Examples are DSORG=PS (physical sequential) and DSORG=PO (partitioned).

*disp* is the status and disposition of the allocation. Examples are DISP=(NEW,CATLG,CATLG) when a new data set is created and DISP=(SHR,KEEP) when an existing data set is retrieved.
System action
FTP continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPBU

Procedure name
logALLOC

EZYFS62I  ID=sessionID ALLOC SMS Storclas=storclass Mgmtclas=mgmtclass Dataclas= dataclass

Explanation
This log entry provides additional information about an allocation for a new MVS data set.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

storclass is the storage class of a new SMS-managed data set.
mgmtclass is the management class of a new SMS-managed data set.
dataclass is the data class of a new SMS-managed data set.

System action
FTP continues.

Operator response
None

System programmer response
None.

Module
EZAFTPBU

Procedure name
logALLOC

EZYFS63I  ID=sessionID ALLOC fails action MVS DSN=dsname

Explanation
This log entry is made by the FTP server when it attempts to allocate an MVS data set and the allocation fails.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.
action indicates whether a new data set was to be created during allocation or an existing data set was to be used. The values for action are the following:

**Create**
Create a new data set.

**Use**
Use an existing data set.

dsnname is the full MVS name of the data set.

**System action**
The MVS data set is not allocated.

**Operator response**
None.

**System programmer response**
Look for log entry messages EZYFS64I and EZYFS65I for additional information.

**Module**
EZAFTPBU

**Procedure name**
logALLOC

| EZYFS64I | ID=sessionID ALLOC SVC 99 RC=rc ERROR=err INFO=info |

**Explanation**
This log entry provides additional information about an unsuccessful allocation for an MVS data set. The information is returned by SVC 99 and is defined in the dynamic allocation chapters of the z/OS MVS Programming: Authorized Assembler Services Guide.

**sessionID** uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

**rc** is the DYNALLOC Return Code when the allocation failed.

**err** is the hexadecimal DYNALLOC Error Reason Code returned when the allocation failed.

**info** is the hexadecimal DYNALLOC Information Reason Code returned when the allocation failed.

**System action**
The MVS data set is not allocated.

**Operator response**
None.

**System programmer response**
See the Error Reason Codes table in the z/OS MVS Programming: Authorized Assembler Services Guide for the meaning and action to be taken for the error. Also, look for the log entry message EZYFS65I for additional information.
Module
EZAFTPBU

Procedure name
logALLOC

**EZYFS65I**  
ID=sessionID ALLOC Message=text

**Explanation**
This log entry provides additional information about an unsuccessful allocation of an MVS data set. 

`sessionID` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

`text` is the text of an extracted message from dynamic allocation. The following are examples of the failure messages that are extracted when the FTP server attempts to write to a data set that is currently being edited:

<table>
<thead>
<tr>
<th>Message</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>EZYFS65I FTP ID=FTP0100001 ALLOC Message=IKJ56225I DATA SET USER33.TEST.S.ALLOC.A1 ALREADY IN USE, TRY LATER+</td>
<td></td>
</tr>
<tr>
<td>EZYFS65I FTP ID=FTP0100001 ALLOC Message=IKJ56225I DATA SET IS ALLOCATED TO ANOTHER JOB OR USER</td>
<td></td>
</tr>
</tbody>
</table>

The IKJ56225I message numbers are data management numbers associated with dynamic allocation.

**System action**
The MVS data is not allocated.

**Operator response**
None.

**System programmer response**
Look at the log entry for message EZYFS64I for additional information.

Module
EZAFTPBU

Procedure name
logALLOC

**EZYFS66I**  
ID=sessionID ALLOC SVC 99 S99ERSN = reason

**Explanation**
This log entry provides additional information about an unsuccessful allocation for an MVS data set. The information is returned by SVC 99 and is defined in the dynamic allocation chapters of the z/OS MVS Programming: Authorized Assembler Services Guide.

`sessionID` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

`reason` is the hexadecimal SMS error reason code S99ERSN.

**System action**
The MVS data set is not allocated.
Operator response
Contact the system programmer.

System programmer response
See z/OS MVS Programming: Authorized Assembler Services Guide for the SMS error reason codes and their meanings. Also, look for the log entry messages EZYFS64I and EZYFS65I for additional information.

Module
EZAFTPGU

Procedure name
logALLOC

EZYFS67I ID=sessionID ALLOC OK action HFS filename=filename

Explanation
This log entry is made by the FTP server when it successfully allocates a z/OS UNIX file.

*sessionID* uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

*action* indicates whether a new file was created during allocation or an existing file was used. The values for action are the following:

**Create**
A new file was created.

**Use**
An existing file was used.

*filename* is the full z/OS UNIX file name.

System action
FTP continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPBU

Procedure name
logALLOC

EZYFS68I ID=sessionID ALLOC fails action HFS filename=filename

Explanation
This log entry is made by the FTP server when it attempts to allocate a z/OS UNIX file and the allocation fails.

*sessionID* uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.
action indicates whether a new file was to be created during allocation or an existing file was to be used. The values for action are the following:

Create
Create a new file.

Use
Use an existing file.

filename is the full z/OS UNIX file name.

System action
The z/OS UNIX file is not allocated.

Operator response
None.

System programmer response
Look for log entry message EZYFS69I for additional information.

Module
EZAFTPBU

Procedure name
logALLOC

EZYFS69I ID=sessionID ALLOC Errno=err Text=text

Explanation
This log entry provides additional information about an unsuccessful allocation for a z/OS UNIX file. sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

err is the errno value that is returned from the fopen() function that attempted the allocation.

text is a description of the error.

System action
FTP continues.

Operator response
None.

System programmer response
See the z/OS XL C/C++ Runtime Library Reference for a description of the failure errno values for the fopen() function. errnos are the UNIX System Services return codes. These return codes are listed and described in the return codes (errnos) information in z/OS UNIX System Services Messages and Codes. If you are not able to correct the problem using the failure descriptions, contact the IBM support center and provide the log entry message number and errno value of the failure.

Module
EZAFTPBU
**Procedure name**

logALLOC

**EZYFS70I**  ID=sessionID DEALL OK action MVS DSN=dsname

**Explanation**

This log entry is made by the FTP server when it successfully deallocates an MVS data set.

`sessionID` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

`action` indicates the action taken when the data set was deallocated. The following are the values for action:

**Release**

Deallocate and keep the data set.

**Delete**

Deallocate and delete the data set.

A data set is usually released at the end of a transfer command. The exception is when a newly created data set is deleted because the transfer failed and the CONDDISP DELETE option was chosen in the FTP.DATA file or by the SITE command. An MVS data set is also deallocated with delete when the DELE command is processed by the server.

`dsname` is the full MVS name of the data set.

**System action**

FTP continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

EZAFTPBU

**Procedure name**

logDEALL

**EZYFS71I**  ID=sessionID DEALL fails action MVS DSN=dsname

**Explanation**

This log entry is made by the FTP server when it attempts to deallocate an MVS data set and the deallocation fails.

`sessionID` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

`action` indicates the action taken when the data set was deallocated. The following are the values for action:

**Release**

Deallocate and keep the data set.

**Delete**

Deallocate and delete the data set.
A data set is usually released at the end of a transfer command. The exception is when a newly created data set
is deleted because the transfer failed and the CONDDISP DELETE option was chosen in the FTP.DATA file or by
the SITE command. An MVS data set is also deallocated with delete when the DELE command is processed by
the server.

dsname is the full MVS name of the data set.

**System action**
The MVS data set is not deallocated.

**Operator response**
None.

**System programmer response**
Look for log entry message EZYFS72I for additional information.

**Module**
EZAFTPBU

**Procedure name**
logDEALL

**EZYFS72I**

<table>
<thead>
<tr>
<th>ID=sessionId</th>
<th>DEALL SVC 99 RC=rc ERROR=err INFO=info</th>
</tr>
</thead>
</table>

**Explanation**
This log entry provides additional information about an unsuccessful deallocation for an MVS data set. The
information is returned by SVC 99 and is defined in the dynamic allocation chapters of the z/OS MVS
Programming: Authorized Assembler Services Guide.

sessionId uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first
message in the log for this session ID.

rc is the decimal DYNALLOC Return Code when the deallocation failed.

er is the decimal DYNALLOC Error Reason Code returned when the deallocation failed.

info is the hexadecimal DYNALLOC Information Reason Code returned when the deallocation failed.

**System action**
The MVS data set is not deallocated.

**Operator response**
None.

**System programmer response**
See the Error Reason Codes table in the z/OS MVS Programming: Authorized Assembler Services Guide for
meaning and action to be taken for the error.

**Module**
EZAFTPBU
Explanation
This log entry provides additional information about an unsuccessful deallocation for an MVS data set. The information is returned by SVC 99 and is defined in the dynamic allocation chapters of the z/OS MVS Programming: Authorized Assembler Services Guide.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

reason is the hexadecimal SMS error reason code S99ERSN.

System action
The MVS data set might not be deallocated correctly.

Operator response
Contact the system programmer.

System programmer response
See z/OS MVS Programming: Authorized Assembler Services Guide for the SMS error reason codes and their meanings. Also, look for the log entry message EZYFS72I for additional information.

Module
EZAFTPBU
System programmer response
The S99INFO information reason code provides additional information about any errors that occurred during the allocation that did not cause the allocation to fail. See the z/OS MVS Programming: Authorized Assembler Services Guide for a description of the S99INFO value.

Module
EZAFTPGU

Procedure name
logALLOC

EZYFS75I ID=sessionID DEALL SVC 99 S99INFO = info

Explanation
This log entry provides additional information about a successful deallocation for an MVS data set. The information is returned by SVC 99 and is defined in the dynamic allocation chapters of the z/OS MVS Programming: Authorized Assembler Services Guide.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

info is the hexadecimal informational reason code S99INFO.

System action
The MVS data set is deallocated. FTP continues.

Operator response
Contact the system programmer.

System programmer response
The S99INFO information reason code provides additional information about any errors that occurred during the deallocation that did not cause the deallocation to fail. See the z/OS MVS Programming: Authorized Assembler Services Guide for a description of the S99INFO value.

Module
EZAFTPBU

Procedure name
logDEALL

EZYFS77I ID=sessionID DEALL OK action HFS filename=filename

Explanation
This log entry is made by the FTP server when it successfully deallocates a z/OS UNIX file.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

action indicates the action taken when the file was deallocated. The following are the values for action:

Release
   Deallocate and keep the file.
Delete
Deallocate and delete the file.

A file is usually released at the end of a transfer command. The exception is when a newly created file is deleted because the transfer failed and the CONDDISP DELETE option was chosen in the FTP.DATA file or by the SITE command. A file is also deallocated and deleted when the DELE command is processed by the server.

filename is the full z/OS UNIX file name.

System action
FTP continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPBU

Procedure name
logDEALL

EZYFS78I  ID=sessionID DEALL fails action HFS filename=filename

Explanation
This log entry is made by the FTP server when it attempts to deallocate a z/OS UNIX file and the deallocation fails.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

action indicates the action taken when the file was deallocated. The following are the values for action:

- Release
  Deallocate and keep the file.

- Delete
  Deallocate and delete the file.

A file is usually released at the end of a transfer command. The exception is when a newly created file is deleted because the transfer failed and the CONDDISP DELETE option was chosen in the FTP.DATA file or by the SITE command. A file is also deallocated and deleted when the DELE command is processed by the server.

filename is the full z/OS UNIX file name.

System action
The file is not deallocated.

Operator response
None.

System programmer response
Look for log entry message EZYFS79I for additional information.
Module
EZAFTPBU

Procedure name
logDEALL

EZYFS79I   ID=sessionID  DEALL  Errno=err  Text=text

Explanation
This log entry provides additional information about an unsuccessful deallocation for a z/OS UNIX file.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

err is the errno value that is returned by the function that was used to deallocate the file.

text is a description of the error.

System action
FTP continues.

Operator response
None.

System programmer response
Look for message EZYFS78I that precedes this message in the log. If the action in the message is Release, the function fclose() was used to deallocate the file. If the action in the message is Delete, the function remove() was used to deallocate the file. The z/OS XL C/C++ Runtime Library Reference describes the failure values for the fclose() and remove() functions. If you are not able to correct the problem using the failure descriptions, contact the IBM Software Support Center and provide the log entry message number, the errno value, and the description of the failure.

Module
EZAFTPBU

Procedure name
logDEALL

EZYFS80I   ID=sessionID  TRANS  Reply=reply

Explanation
This log entry is the last entry made by the FTP server for a data transfer command process. It logs the reply that is sent to the client to report the completion of the data transfer.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

reply is data that begins with a 3-digit reply code followed by the text for the reply. The first digit of the 3-digit reply indicates success or failure defined as follows:

2  Positive Completion reply. The requested action has been successfully completed.

4  Transient Negative Completion reply. The command was not accepted and the requested action did not take place, but the error condition is temporary and the action might be requested again.
Permanent Negative Completion reply. The command was not accepted and the requested action did not take place.

For more information about FTP reply codes, see RFC 959. See Appendix A, “Related protocol specifications,” on page 1055 for information about accessing RFCs.

System action
FTP continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPBU

Procedure name
logTRANS

Explanation
This log entry is the first entry made by the FTP server for a data transfer command process for an MVS data set. It logs the name of the MVS data set that the server is sending from or receiving into.

`sessionID` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this `sessionID`.

`dsname` is the full MVS name of the data set.

System action
FTP continues.

Operator response
None

System programmer response
None.

Module
EZAFTPBU

Procedure name
logTRANS

Explanation
This log entry is the first entry made by the FTP server for a data transfer command process for an HFS file. It logs the name of the file that the server is sending from or receiving into.

`sessionID` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this `sessionID`.

`filename` is the full HFS name of the file.

System action
FTP continues.

Operator response
None

System programmer response
None.

Module
EZAFTPBU

Procedure name
logTRANS

Explanation
This log entry is the first entry made by the FTP server for a data transfer command process for an MVS data set. It logs the name of the MVS data set that the server is sending from or receiving into.

`sessionID` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this `sessionID`.

`dsname` is the full MVS name of the data set.
Explanation
This log entry is the first entry made by the FTP server for a data transfer command process for a z/OS UNIX file. It logs the name of the z/OS UNIX file that the server is sending from or receiving into.

`sessionID` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

`filename` is the full z/OS UNIX file name.

System action
FTP continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPBU

Procedure name
logTRANS

**EZYFS83I**    ID=`sessionID` TRANS=Stru=`stru` Mode=`mode` Type=`type` Input=`bytesIn`

Explanation
This log entry contains information about an input data transfer from the perspective of the FTP server. Processing for a STOR, STOU, or APPE command results in an input data transfer.

`sessionID` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

`stru` is the data structure for the transfer. The following are the values for `stru`:

- **F**  
  file structure
- **R**  
  record structure

`mode` is the transmission mode. The following are the values for `mode`:

- **S**  
  stream mode
- **B**  
  block mode
- **C**  
  compressed mode

`type` is the data type. The following are the values for `type`:

- **A**  
  ASCII
- **E**  
  EBCDIC
bytesIn is the number of bytes transferred on the data connection for this command. If a number is greater than or equal to one gigabyte (that is, 1,073,741,824 bytes), the number is displayed as a number of gigabytes rounded to the nearest hundredth of a gigabyte. Some examples are:

1,073,741,824 bytes is displayed as 1.00 GB
2,147,483,648 bytes is displayed as 2.00 GB
2,415,919,104 bytes is displayed as 2.25 GB

System action
FTP continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPBU

Procedure name
logTRANS

EZYFS84I ID=sessionID TRANS Stru=stru Mode=mode Type=type Output=bytesOut

Explanation
This log entry contains information about an output data transfer from the perspective of the FTP server. Processing for a RETR command results in an output data transfer.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

stru is the data structure for the transfer. The following are the values for stru:
F file structure
R record structure

mode is the transmission mode. The following are the values for mode:
S stream mode
B block mode
C compressed mode

type is the data type. The following are the values for type:
bytesOut is the number of bytes transferred on the data connection for this command. If a number is greater than or equal to one gigabyte (that is, 1,073,741,824 bytes), the number is displayed as a number of gigabytes rounded to the nearest hundredth of a gigabyte. Some examples are:

- 1,073,741,824 bytes is displayed as 1.00 GB
- 2,147,483,648 bytes is displayed as 2.00 GB
- 2,415,919,104 bytes is displayed as 2.25 GB

**Explanation**

This log entry is the first entry made by the FTP server for a data transfer that returns the output from a job when the FTP filetype is JES.

`sessionId` uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

`jobid` is the job identifier associated with the JES job.

`ddname` is the ddname associated with spool file.

`spool` is the name of the spool file.

This message is followed in the log by a message EZYFS84I and EZYFS80I.
Operator response
None.

System programmer response
None.

Module
EZAFTPBU

Procedure name
logTRANS

EZYFS86I  ID=sessionID TRANS Confidence=confidence_level

Explanation
This log entry is written by the FTP server to report the confidence level in the successful completion of a file transfer when CHKConfidence TRUE is specified or defaulted to in the server FTP.DATA data set. The preceding message EZYFS81I or EZYFS82I contains the name of the data set or file being stored by the server.

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

confidence_level is the level of confidence that the FTP server has in the completion of the transfer. Values for confidence_level are:

High
While successful completion of the transfer cannot be guaranteed, the FTP server did not detect a reason to doubt completion of the transfer.

NoEOF
The FTP server detected a missing EOF marker in a STRU R or MODE B or C inbound file. This level is reported only when no other problem is detected.

Low
The FTP server detected a problem with completion of the transfer such as the failure of the client to respond after the transfer or another reported error. Low overrides NoEOF if both conditions are present.

Unknown
This confidence level is reported when the direction of the file transfer was outbound. If an error occurs while shutting down the data connection, then confidence in an outbound transfer will be reported as Low. In all other cases, the confidence level is reported as Unknown because the server cannot perform any other checks on an outbound transfer.

System action
FTP continues.

Operator response
Check to see whether the transfer completed successfully and try again, if necessary.

System programmer response
None.

Module
EZAFTPBU
Procedure name
logTRANS()

**EZYFS91I**  ID=sessionID SUBMIT JES JobID=jobid

**Explanation**
This log entry is made by the FTP server for a data transfer that submitted a job when the FTP filetype is JES. *sessionID* uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

*jobid* is the job identifier associated with the JES job. The job identifier is used later by the client end user to retrieve the output of the submitted job.

**System action**
FTP continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPBU

**Procedure name**
logSUBMIT

**EZYFS92I**  ID=sessionID QUERY SQL filename=filename

**Explanation**
This log entry is the first entry made for a data transfer of a report requested when the FTP filetype is SQL. *sessionID* uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

*filename* is the full z/OS UNIX file name or MVS data set name that contains the select statement used to make the query.

This message is followed in the log by messages EZYFS84I and EZYFS80I.

**System action**
FTP continues.

**Operator response**
None.

**System programmer response**
None.
Module
EZAFTPBU

Procedure name
logQUERY

EZYFS95I  ID=sessionID  ABEND  action  COMP_CODE=code  DSN=dsname

Explanation
This log entry records the system completion code when an ABEND occurs during an event for an MVS data set. 

sessionID uniquely identifies the FTP session between a client and a server. Message EZYFS50I is the first message in the log for this session ID.

action is the name of the operation in progress when the failure occurred. The values are READ, WRITE, and CLOSE.

code is the ABEND completion code.

dsname is the full MVS name of the data set.

The following is an example of an ABEND message that occurs when the receiving data set is too small to hold the data that is transferred:

EZYFS95I  ID=FTPDJG100001  ABEND  WRITE  COMP_CODE=D37-04  DSN=USER33.TOOBIG

System action
FTP continues.

Operator response
None.

System programmer response
See the z/OS MVS System Codes for the system programmer response for the completion code in this message.

Module
EZAFTPBU

Procedure name
logABEND

EZYFT01I  Unable to open message catalog 'message catalog'. error text. Using FTP's default messages.

Explanation
An attempt was made to open the FTP server's message catalog (named 'ftpdmsg.cat') in the directory determined by the NLSPATH and LANG environment variables, but the catalog could not be opened for the reason explained in error text.

System action
Processing continues. Default messages will be used.

Operator response
None.
**System programmer response**

If a message catalog is required, stop the server, correct the problem as indicated by the error text, and restart the server.

**Module**

EZAFTPDM

**EZYFT02E**  Filename exceeds maximum valid length of max.

**Explanation**

The FTP server was processing a statement in the FTP.DATA file that requires a file name, such as SBDATACONN or SOCKSCONFIGFILE. The file name specified is longer than max characters. This message is preceded by message EZYFT46E which locates the error in the FTP.DATA file. The keyword is ignored and the FTP server continues with the next keyword.

**System action**

FTP continues.

**Operator response**

None.

**System programmer response**

Correct the file name provided on the statement in the FTP.DATA file. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

**Module**

EZAFTPEP

**EZYFT04E**  Invalid syntax. The syntax must be: (value1,value2)

**Explanation**

The value provided for a statement in the FTP.DATA file does not have valid syntax. This message is preceded by message EZYFT46E which locates the error in the FTP.DATA file.

**System action**

The statement is ignored. Processing continues.

**System action**

Processing continues.

**Operator response**

Contact the System programmer with both messages to have the FTP.DATA file corrected.

**System programmer response**

Use the EZYFT46E message to locate the error in the FTP.DATA file, and correct the error described by this message.

**Module**

EZAFTPEP
EZYFT05I Volume_serial_list is not a valid VOLUME parameter.

Explanation
While processing the VOLUME statement in the FTP.DATA file, a syntax error was detected in volume_serial_list.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues.

Operator response
Contact the System programmer with the error message to correct the FTP.DATA file.

System programmer response
Correct the FTP.DATA file to contain the correct value for volume_serial_list. See the z/OS Communications Server: IP Configuration Reference for more information about the parameters of the FTP.DATA file.

Module
EZAFTPEP

Procedure name
read_ftpdata()

EZYFT06I Using internal messages for replies because there is no reply catalog available.

Explanation
FTP was unable to locate or open a reply catalog (ftpdrply.cat). The reply texts contained within the FTP modules will be used for FTP replies.

System action
Processing continues.

Operator response
None.

System programmer response
If an external reply catalog is required, either place a copy of ftpdrply.cat in the 'C' subdirectory of your nlspath, or use the REPLYLANGUAGE parameter in the FTP.DATA file to establish one or more reply catalogs to be used. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

Module
EZAFTPEP

EZYFT07E UMASK value 'value' not a valid octal value

Explanation
The value, value, specified for the UMASK keyword in the FTP.DATA configuration file was not a valid 3 character octal number. The value of the UMASK keyword must be a 3 character octal number in the range of 000 - 777.
**System action**
The keyword is ignored. Processing continues with the next keyword.

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

**System programmer response**
Correct the value specified for UMASK in the FTP.DATA file. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

**Module**
EZAFTPEP

**EZYFT08W** Unable to get port number from etc.services : error

**Explanation**
The FTP server was unable to retrieve its port number from the etc.services file. error is the error message returned by the C run-time library for the failing getservbyname() routine.

**System action**
Processing continues. The FTP server will attempt to use either the value of the PORT start option, if specified, or the default port of port 21.

**Operator response**
None.

**System programmer response**
Verify the search path to the ETC.SERVICES file and verify that an entry exists in the appropriate ETC.SERVICES file for FTP. If the PORT start option is specified, or if PORT 21 is an acceptable default port, this message may be ignored. Otherwise, add an entry for the ftp server to the appropriate etc.services file. See the z/OS Communications Server: IP Configuration Reference information about configuring the etc.services file for the FTP server.

**Module**
EZAFTPDM

**EZYFT09I** system information for nodename: sysname version version release release (machine)

**Explanation**
This is an informational message describing the MVS host which the ftp server is running on. nodename is the name of the node within an implementation-specified communication network. sysname is the name of the implementation of the operating system. version is the version level of the operating system. release is the release level of the operating system. machine is the name of the hardware type the system is running on.

**System action**
Processing continues.

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

**Module**
EZAFTPDM

**EZYFT12E**  socket error : *error*

**Explanation**
The ftp server encountered an error while attempting to create the socket for the control port. *error* is the C run-time library error message returned for the failing socket() call.

**System action**
If TCP/IP is not available (resource temporarily unavailable), the FTP server will try again in 60 seconds. If any other error has occurred, the FTP server is ended with exit code 0012.

**Operator response**
None.

**System programmer response**
If *error* indicates that a resource is temporarily unavailable, ensure that TCP/IP has been started. Otherwise, correct the error indicated by *error* and restart the FTP server.

**Module**
EZAFTPSK

**EZYFT13E**  bind error : *error*

**Explanation**
The ftp server encountered an error while attempting to bind the socket for the control port. *error* is the C run-time library error message returned for the failing bind() call.

**System action**
The FTP server is ended.

**Operator response**
None.

**System programmer response**
Correct the error indicated by *error*.

**Module**
EZAFTPSD

**EZYFT14E**  listen error : *error*

**Explanation**
The ftp server encountered an error while attempting to listen on the socket for the control port. *error* is the C run-time library error message returned for the failing listen() call.
**System action**
The FTP server is ended.

**Operator response**
None.

**System programmer response**
Correct the error indicated by `error`.

**Module**
EZAFTPSK

**EZYFT15E**  
**selectex error : error**

**Explanation**
The ftp server encountered an error while attempting to setup the control port. `error` is the C run-time library error message returned for the failing selectex() call.

**System action**
The FTP server is ended.

**Operator response**
None.

**System programmer response**
Correct the error indicated by `error`.

**Module**
EZAFTPSK

**EZYFT16E**  
**accept error : error**

**Explanation**
The ftp server encountered an error while attempting to setup the control port. `error` is the C run-time library error message returned for the failing accept() call.

**System action**
The FTP server is ended.

**Operator response**
None.

**System programmer response**
Correct the error indicated by `error`.

**Module**
EZAFTPSK

**EZYFT17E**  
**getsockname error : error**
**Explanation**
The ftp server encountered an error while attempting to setup the control port. *error* is the C run-time library error message returned for the failing getsockname() call.

**System action**
The FTP server is ended.

**Operator response**
None.

**System programmer response**
Correct the error indicated by *error*.

**Module**
EZAFTPSK

**EZYFT18I** Using catalog 'catalog' for FTP messages.

**Explanation**
The messages issued by FTP server or client is retrieved from the message catalog in *catalog*.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPDM and EZAFTPCY

**EZYFT19E** Unrecognized code page name: 'name'

**Explanation**
A CTRLCONN statement in the FTP.DATA file has an invalid value. Valid values include '7bit' or name of a code set that is recognized as part of the code set converters used by z/OS UNIX System Services. This message follows message EZYFT46E that provides the location of the error.

**System action**
The statement is ignored. Processing continues.

**Operator response**
None.
System programmer response
Use EZYFT46E to locate the error within the FTP.DATA file. See "Code Set Converters Supplied" in z/OS XL C/C++ Programming Guide for a list of recognized code pages. The code page name must be entered exactly as it appears in the list.

Module
EZAFTPEP

EZYFT20E Invalid parameter length. Maximum length for parameter 1 is max1. Maximum length for parameter 2 is max2.

Explanation
An invalid parameter length was encountered while processing a statement in the FTP.DATA file. This message follows message EZYFT46E which provides the location of the error.

System action
The line in FTP.DATA containing the error is ignored. Processing continues.

Operator response
Contact the System programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

Module
EZAFTPEP

EZYFT21I Using catalog 'file' for FTP replies.

Explanation
The FTP replies sent to the client by the FTP server will be retrieved from the catalog in file.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPEP

EZYFT22E Unexpected end of file.
Explanation
The FTP server was attempting to load a translation table, but an unexpected end of file occurred while attempting to read the file. This message will be preceded by message EZY2689E which will identify the file being read at the time of the error.

System action
The FTP server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the internally provided translation tables are used.

Operator response
None.

System programmer response
Determine why end of file was encountered. It might be necessary to rebuild the TCPXLBIN file using the CONVXLCATE command.

Module
EZAFTPDX

EZYFT23E No conversion available between CTRLCONN parameter ('parameter') and the FTP server's code page ('codepage').

Explanation
The FTP server was processing a CTRLCONN parameter in the FTP.DATA file. The parameter value (parameter) does not indicate a valid ASCII choice for establishing translate tables for FTP's control connection. Valid values include '7bit' (specified in lower, upper, or mixed case) and single-byte ASCII code set names that are recognized by the iconv function.

System action
The CTRLCONN statement is ignored. The FTP server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the internally provided translation tables are used.

Operator response
None.

System programmer response
If an iconv-generated translate table is required for the control connection, correct the CTRLCONN statement and restart the FTP server. See z/OS XL C/C++ Programming Guide for information about supported code set converters and the code set names recognized by iconv.

Note: If the FTP server is running in a double-byte code page, the CTRLCONN parameter cannot be used to establish translate tables for the control connection.

Module
EZAFTPDXN

EZYFT24E Unable to set up conversion between 'page1' and 'page2'.

624 z/OS Communications Server: IP Messages: Volume 3 (EZY)
Explanation
The FTP server was processing either a CTRLCONN or SBDATACONN parameter in the FTP.DATA file. A code set converter was successfully opened, but an error occurred while attempting to set up single-byte translate tables using the indicated code sets page1 and page2.

System action
The FTP server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the internally provided translation tables are used.

Operator response
None.

System programmer response
Valid code sets are listed in the z/OS XL C/C++ Programming Guide in the section about the code set converters supplied. If the requested code sets are double-byte code sets, the use of CTRLCONN or SBDATACONN to establish translate tables is not supported, and the statement should be removed from the FTP.DATA file. FTP validates the results of ICONV and returns an error if there are missing code points. If the code sets are valid single-byte code sets, an internal error has occurred. Contact the IBM Software Support Center with this message and the FTP trace output, if it is available.

Module
EZAFTPNX

EZYFT25I Using file for FTP translation tables for the control connection.

Explanation
file is the name of the file that was used to set up the translate tables for the control connection.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPNX

EZYFT26I Using 7-bit conversion derived from 'codeset1' and 'codeset2' for the control connection.

Explanation
The FTP server has processed a CTRLCONN statement in the FTP.DATA file. The statement indicated a 7-bit table was required. The code set codeset1 was used for the ASCII code set, and codeset2 was used for the EBCDIC code set, but only 7-bit translations appear in the translate table.

System action
Processing continues.
Operator response
None.

System programmer response
None.

Module
EZAFTPNX

EYZFT27I  Using conversion between 'codeset1' and 'codeset2' for the control connection.

Explanation
The FTP server has processed a CTRLCONN statement in the FTP.DATA file. The translate table for the control connection was built using code set codeset1 for the ASCII code set, and codeset2 for the EBCDIC code set.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPNX

EYZFT28W  Unable to use iconv to establish default translate tables for the control connection. Using internal tables.

Explanation
FTP was unable to use iconv to build 7-bit translate tables based on ISO8859-1 and the current host code set. Internal 7-bit tables will be used. This will occur if the FTP server is running in a double-byte code page and no other translate tables were found in the search order.

System action
Processing continues.

Operator response
None.

System programmer response
If the FTP server is running in a single-byte code page, an internal error has occurred. Contact the IBM support center with this error message and the output from the FTP server trace.

Module
EZAFTPNX

EYZFT29I  Using conversion between 'codeset1' and 'codeset2' for the data connection.
Explanation
The FTP server has processed a SBDATACONN statement in the FTP.DATA file. The translate table for the data connection was built using code set codeset1 for the EBCDIC code set, and codeset2 for the ASCII code set.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPNX

EZYFT30E  No conversion available between SBDATACONN parameters: 'parm1' and 'parm2'.

Explanation
The FTP server was processing an SBDATACONN parameter in the FTP.DATA file. There is no supported code set converter for the code sets parm1 and parm2. Valid code sets for the first SBDATACONN parameter include the single-byte EBCDIC code set names recognized by the iconv function; valid code sets for the second parameter include the single-byte ASCII code set names.

System action
The SBDATACONN statement is ignored. The FTP server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the translation tables set up for the control connection will also be used for the data connection.

Operator response
None.

System programmer response
If an iconv-generated translate table is required for the data connection, correct the SBDATACONN statement and restart the FTP server. See z/OS XL C/C++ Programming Guide for information about supported code set converters and the code set names recognized by iconv.

Module
EZAFTPNX

EZYFT31I  Using file for FTP translation tables for the data connection.

Explanation
file is either the file name or the ddname for the file that was used to build the translate tables for the data connection.

System action
Processing continues.
Using the same translate tables for the control and data connections.

Explanation
The FTP server was not able to set up translate tables following the search order for the data connection: DD: SYSFTSX, SBDATACONN or Xlate in FTP.DATA, TCPXLBIN file. The same translate table established for the control connection will also be used for the data connection.

System action
Processing continues.

Unable to open DDNAME 'ddname' for the data connection: reason

Explanation
The FTP server attempted to open ddname for data connection translate tables, but the open failed for the specified reason.

System action
The FTP server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the translate tables set up from the control connection will also be used for the data connection.

Operator response
If ddname is the required file, correct the error specified by reason. If the required translation table is further in the search order, no action is necessary.

System programmer response
If ddname is the required file, correct the error specified by reason. If the required translation table is further in the search order, no action is necessary.
**EZYFT34W**  Sigaction for signal failed : error (errno/errnojr)

**Explanation**
The FTP server encountered an error while attempting to setup the signal handler for the signal specified by `signal`. If the signal handler is not correctly enabled, the server will continue processing, but certain functions controlled by the failing signal will not function properly. Functions controlled by the signals are:

**SIGABND**
handler controls error reporting and cleanup functions when an abend occurs. If sigaction fails for SIGABND and an abend occurs, trace information about the abend will be lost and certain resources might not be properly cleaned up.

**SIGCHLD**
handler controls cleanup of zombie processes when a client connection is ended. If sigaction fails for SIGCHLD, zombie processes will not be cleaned up when a client connection is ended.

**SIGTERM**
handler controls cleanup of resources during termination. If sigaction fails for SIGTERM, the FTP server will not be cleanly terminated when an MVS operator STOP command is issued or when the server process is killed.

**SIGURG**
handler controls the processing of Out of Band data, such as the ABOR subcommand. If sigaction fails for SIGURG, OOB data will not be received or processed by the server.

`error` is the error returned by the C run-time library for the failing sigaction() call.

`errno` is the 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 UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing continues; however, the functions controlled by the failing signal will not function properly.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPDM, EZAFTPRX, EZAFTPHC

**Module**
EZYFT40E  FTP server initialization failed - error

**Explanation**
The FTP server was unable to successfully complete initialization. `error` is the error returned by the C Runtime Library for the failing function.

**System action**
The FTP server is ended.
Operator response
None.

System programmer response
Correct the error indicated by error.

Module
EZAFTPD

**EZYFT41I**  Server-FTP: process id *pid*, server job name *jobname*

Explanation
This is an information message indicating the process ID (*pid*) and the server job name (*jobname*) of the FTP server after initialization has completed. The jobname can be used in an MVS operator STOP command to stop the FTP server, or in an MVS operator MODIFY command to control tracing for the ftp server. The process ID can be used in an OMVS "kill" command to terminate the server. The process ID can also be used to identify trace entries for the FTP server in the SYSLOGD output files.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPSK

**EZYFT42E**  Unable to use file 'file' for translate tables for the data connection.

Explanation
The FTP server was unable to load the translate tables from the file *file* that was specified by an SBDATACONN or XLATE statement in the FTP.DATA file. If FTP tracing was enabled during initialization, additional messages will precede this one with more specific detail about the error encountered. Possible errors include an invalid header record or incorrect file length. (The file must contain exactly 768 bytes of data.)

System action
The SBDATACONN or XLATE statement is ignored. The FTP Server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the translation tables set up for the control connection will also be used for the data connection.

Operator response
None.

System programmer response
If file specified by the SBDATACONN or XLATE statement is required, determine why it was not usable, correct the problem, and stop and restart the FTP server.
EZYFT43E  Unrecognized LOADDBCSTABLES parameter: parameter. Parameter ignored.

Explanation
The LOADDBCSTABLES statement in the TCPIP.DATA file contains an invalid parameter (parameter).

System action
The parameter is ignored by FTP. The rest of the parameters on the statement are processed. Processing continues.

Operator response
None.

System programmer response
Correct the LOADDBCSTABLES statement in the TCPIP.DATA file. If the needed keywords were not already present in the LOADDBCSTABLES statement, stop and restart the FTP server.

EZYFT44E  Translate table is too small.

Explanation
The FTP server encountered an error while processing a TCPXLBIN file. The file must be exactly 768 bytes in length.

System action
The FTP server continues through the translate table search order, attempting to find a valid translation table.

Operator response
None.

System programmer response
Verify that the TCPXLBIN file being used has the correct format.

EZYFT45E  Translate table is too large.

Explanation
The FTP server encountered an error while processing a TCPXLBIN file. The file must be exactly 768 bytes in length.

This error will occur if a valid translate table is copied and line control characters are added for each line of the table. (That is, the size of the table is now 771 bytes.)
System action
The FTP server continues through the translate table search order, attempting to find a valid translation table.

Operator response
None.

System programmer response
Verify that the TCPXLBIN file being used has the correct format.

Module
EZAFTPDY

**EZYFT46E** Error in name file: line *line_number* near column *column_number*

**Explanation**
An error was detected in the FTP.DATA file. *name* is the name of the file being used as the FTP.DATA file. This will be either "DD:SYSFTPD", indicating that the FTP.DATA file is the one specified by the SYSFTPD DD statement, or it will be the actual file name if the FTP.DATA file was not the one specified on the SYSFTPD DD statement. *line_number* is the number of the line in the FTP.DATA file which contains the error. *column_number* is the approximate location of the error within the line.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

Operator response
This message should be followed by another message which describes the error. Contact the System programmer with both messages to have the FTP.DATA file corrected.

System programmer response
This message should be followed by another message which describes the error. Correct the error described by the second message.

Module
EZAFTPEP

**EZYFT47I** ftp_data file, line *line_number*: Ignoring keyword "*keyword*".

**Explanation**
While processing the FTP.DATA file, the FTP server encountered a keyword that was valid for another FTP server or the FTP client, but that is unsupported by this server. Or, the FTP server is ignoring *keyword* because it is inconsistent with another keyword in FTP.DATA.

In the message text:

*ftp_data*
   The name of the file being used as the FTP.DATA file.

*line_number*
   The number of the line in the FTP.DATA file that contains the error.

*keyword*
   The FTP configuration statement that is being ignored.
These keywords are ignored when the default or explicit ANONYMOUSLEVEL value does not support the keyword:

- ANONYMOUSHFSFILEMODE
- ANONYMOUSHFSDIRMODE
- ANONYMOUSFILETYPEJES
- ANONYMOUSFILETYPESEQ
- ANONYMOUSFILETYPESQL
- EMAILADDRCHECK
- ANONYMOUSFILEACCESS

These keywords are ignored when ANONYMOUS is not explicitly defined in FTP.DATA.

- ANONYMOUSHFSFILEMODE
- ANONYMOUSHFSDIRMODE
- ANONYMOUSFILETYPEJES
- ANONYMOUSFILETYPESEQ
- ANONYMOUSFILETYPESQL
- EMAILADDRCHECK
- ANONYMOUSFILEACCESS
- ANONYMOUSLEVEL

These keywords are ignored when the TLSMECHANISM ATTLS parameter is explicitly defined in FTP.DATA.

- KEYRING
- CIPHERSUITE
- TLSTIMEOUT

**System action**

The FTP server ignores the keyword.

**Operator response**

Contact the system programmer.

**System programmer response**

If the FTP.DATA file is used only by the FTP server, and is not shared with another server or client that needs the keyword, remove the keyword from the FTP.DATA file. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

**Module**

EZAFTPEP

**EZYFT48E**  Error opening STDOUT or STDERR to /dev/null : error

**Explanation**

During FTP server initialization, the FTP server attempted to open STDOUT and STDERR to /dev/null. The open was unsuccessful. error is the error message returned by the C run-time library.

**System action**

The FTP server continues; however, without the STDOUT or STDIN file the LIST and NLST commands will be rejected for z/OS UNIX files.
Operator response
Contact the system programmer.

System programmer response
Correct the error indicated by error.

Module
EZAFTPDM
EZYFT49I
Statement statement parameter must be one of the following: list_of_supported_parameters

Explanation
A statement in FTP.DATA is coded with a parameter that is not supported.
statement is the statement that is coded incorrectly.
list_of_supported_parameters is a list of parameters that are supported for this statement.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

Operator response
Contact the system programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Change statement in FTP.DATA file to specify a supported parameter. See z/OS Communications Server: IP Configuration Reference for information about supported parameters for statements coded in FTP.DATA.

Module
EZAFTPEP

Procedure name
read_ftpdata()

EZYFT50E
JESRECFM value must be one of: F, V, or *.

Explanation
While processing the FTP.DATA file, the FTP server encountered the JESRECFM parameter, but the value specified for the parameter was not a valid value. The value must be one of the values listed in the message.

System action
The line containing the JESRECFM parameter is ignored. Processing continues with the next line in the file.

Operator response
Contact the system programmer with the error message.

System programmer response
Correct the value of the JESRECFM parameter in the FTP.DATA file to be a valid record format. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.
**EZYFT53E**  Unable to execute FTP server load module *module* : *reason*

**Explanation**
The FTP server issued an execv() for the load module named *module* but the execv() failed for the specified reason. The FTP server is unable to process incoming connections.

**System action**
The FTP session for the connecting client is ended. The FTP daemon remains active, awaiting client connections.

**Operator response**
Contact the system programmer.

**System programmer response**
Correct the error indicated by *reason*.

---

**EZYFT55E**  STARTDirectory value must be either MVS or HFS.

**Explanation**
While processing the FTP.DATA file, the FTP Server encountered the STARTDirectory parameter, but the value specified for the parameter was not a valid value. The value must be one of the values listed in the message.

**System action**
The line containing the STARTDirectory parameter is ignored. Processing continues with the next line in the file.

**Operator response**
Contact the system programmer with the error message.

**System programmer response**
Correct the value of the STARTDirectory parameter in the FTP.DATA file to be a valid column heading. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

---

**EZYFT58W**  SMF STD overrides *smfxxxx* on line *line*

**Explanation**
While processing the FTP.DATA file, the FTP Server encountered both SMF STD and SMFxxxx. If SMF STD is coded, none of the SMFxxxx options can be coded.

**System action**
The line containing the smfxxxx option is ignored. Processing continues with the next line in the file.
Operator response
Contact the system programmer with the error message.

System programmer response
If using SMF STD, remove all SMFxxxx options from FTP.DATA. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

Module
EZAFTPEP

EZYFT59I FTP shutdown complete.

Explanation
The FTP daemon has been terminated either by an MVS operator STOP command, or by an OMVS kill command. No new FTP sessions will be accepted.

System action
The FTP daemon ends. Any FTP sessions currently active are not effected.

Operator response
None.

System programmer response
None.

Module
EZAFTPDH

EZYFT60I Both CCXLATE and CTRLCONN were specified. CCXLATE will be ignored.

Explanation
The FTP daemon has encountered both CCXLATE and CTRLCONN keywords in FTP.DATA. Each is used to specify an initial translate table for the control connection, but CTRLCONN is preferred. When both keywords are present in FTP.DATA, the CTRLCONN value will be used for the server configuration and the CCXLATE value will be ignored.

System action
FTP continues.

Operator response
None.

System programmer response
To avoid this message, remove either the CCXLATE or CTRLCONN statement from the FTP.DATA file.

Module
EZAFTPEP

EZYFT61I Both XLATE and SBDATACONN were specified. XLATE will be ignored.
Explanation
The FTP daemon has encountered both XLATE and SBDATACONN keywords in FTP.DATA. Each is used to specify the initial translate table for the data connection, but SBDATACONN is preferred. When both keywords are present in FTP.DATA, the XLATE value will be ignored. The SBDATACONN value will be used for the server configuration unless DD:SYSFTSX has been defined.

System action
FTP continues.

Operator response
None.

System programmer response
To avoid this message, remove either the XLATE or SBDATACONN statement from the FTP.DATA file.

Module
EZAFTPEP

EZYFT62E Unable to use file 'file' for translate tables for the control connection.

Explanation
The FTP server was unable to load the translate tables from the file file that was specified by a CCXLATE statement in the FTP.DATA file. If FTP tracing was enabled during initialization, additional messages will precede this one with more specific detail about the error encountered. Possible errors include an invalid header record or incorrect file length. (The file must contain exactly 768 bytes of data.)

System action
The CCXLATE statement is ignored. The FTP Server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, default 7-bit translate tables will be built.

Operator response
None.

System programmer response
If file specified by the CCXLATE statement is required, determine why it was not usable, correct the problem, and stop and restart the FTP server.

Module
EZAFTPNX

EZYFT63E Unable to use CCXLATE configuration parameter name

Explanation
The FTP server was unable either to locate or to use the file determined by the CCXLATE statement in the FTP.DATA file. (The file name is hlq.name.TCPXLBIN unless this name was superseded by an environment variable called _FTPXLATE_name that defines a file name.) This message will be preceded or followed by another message with additional information. If FTP tracing was enabled during initialization, additional messages will precede this one with more specific detail about the error encountered. Possible errors include an invalid header record or incorrect file length. (The file must contain exactly 768 bytes of data.)
System action
The CCXLATE statement is ignored. The FTP Server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, default 7-bit translate tables will be built.

Operator response
None.

System programmer response
If file specified by the CCXLATE statement is required, determine why it was not usable, correct the problem, and stop and restart the FTP server.

Module
EZAFTPNX

EZYFT64E Unable to use XLATE configuration parameter name

Explanation
The FTP server was unable either to locate or to use the file determined by the XLATE statement in the FTP.DATA file. (The file name is hlq.name.TCPXLBIN unless this name was superseded by an environment variable called _FTPXLATE_name that defines a file name.) This message will be preceded or followed by another message with additional information. If FTP tracing was enabled during initialization, additional messages will precede this one with more specific detail about the error encountered. Possible errors include an invalid header record or incorrect file length. (The file must contain exactly 768 bytes of data.)

System action
The XLATE statement is ignored. The FTP Server will continue through the search order of translation tables attempting to find a good translation table. If no translation tables are found, the translate tables established for the control connection will also be used for the data connection.

Operator response
None.

System programmer response
If file specified by the XLATE statement is required, determine why it was not usable, correct the problem, and stop and restart the FTP server.

Module
EZAFTPNX

EZYFT65E Filename defined by environment variable variable exceeds the maximum length of length

Explanation
The FTP.DATA file contains a 'CCLXATE name' (or 'XLATE name') statement to specify translate tables for the control (or data) connection, but the _FTPXLATE_name environment variable defines a file name that is too long and cannot be used for the translate tables file. This message will be followed by another message that displays the CCXLATE (or XLATE) parameter from FTP.DATA.
System action
The CCXLATE (or XLATE) statement is ignored. The FTP Server will continue through the search order of translation tables attempting to find a good translation table.

Operator response
None.

System programmer response
If tracing was active when the FTP server was initialized, all existing environment variables will be displayed in the trace. Ensure that the _FTPXLATE_name variable is correct and that it specifies a file name that is not longer than length characters.

Module
EZAFTPNX

EZYFT67E UCOUNT value value is not valid. Value must be P, or an integer from 1 through 59.

Explanation
While processing the FTP.DATA file, the server encountered the UCOUNT parameter with a value that was not a numeric value from 1 through 59, nor was it the letter P. Only values 1 to 59, or the letter P, are valid values for the UCOUNT parameter.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues.

Operator response
Contact the System programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain a correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

Module
EZAFTPEP

Procedure name
read_ftpdata()

EZYFT68I ANONYMOUSFILEACCESS value must be MVS, HFS, or BOTH

Explanation
While processing the FTP.DATA file, the ANONYMOUSFILEACCESS parameter was encountered with a value other than MVS, z/OS UNIX file system, or BOTH. These are the only valid values for ANONYMOUSFILEACCESS.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

Operator response
Contact the System programmer with the error message to have the statement in the FTP.DATA file corrected.
System programmer response
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

Module
EZAFTPEP

Procedure name
read_ftpdata()

EZYFT69I  parameter value value is not valid -- parameter value must be three octal digits

Explanation
While processing the FTP.DATA file, a parameter was encountered that requires a 3-digit octal value. The actual value specified, value, is not three octal digits. Parameter is the parameter in error.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

Operator response
Contact the System programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information about the parameters of the FTP.DATA file.

Module
EZAFTPEP

Procedure name
read_ftpdata()

EZYFT70I  SERVER-FTP: ACCEPTING CONNECTIONS

Explanation
The FTP Server is accepting connections from all known common INET (CINET) stacks. This message appears when the MVS operator has just started a CINET stack, and the FTP server has recognized the new stack as well as other CINET stacks active at the time.

System action
Processing continues.

Operator response
None.

System programmer response
None.
EZYFT71E  operation failed in function - error

Explanation
The FTP server encountered a severe error.
operation is the operation in progress.
function is the C Library function that failed and returned an error.
error is the C run-time library error message for the failure. For more information, see z/OS Language Environment Runtime Messages.

System action
The FTP server ends.

Operator response
Contact the system programmer.

System programmer response
Correct the error indicated by error. If you are unable to resolve the problem, contact the IBM Software Support Center and report the contents of this message.

Module
EZAFTPSD

Procedure name
N/A

EZYFT72I  FTP SERVER SUSPENDING INCOMING CONNECTIONS

Explanation
The FTP server stopped accepting incoming connections because a common INET (CINET) stack just started. While the FTP server is reinitializing, no connections can be accepted. This condition is temporary. Message EZYFT70I will follow when the FTP server is accepting connections again.

System action
Processing continues.

Operator response
Wait for message EZYFT70I before attempting to connect to the FTP server.

System programmer response
None.

Module
EZAFTPSD

EZYFT73I  ANONYMOUS LOGINS NOT ALLOWED -- REASON IS reason code
Explanation

The FTP server found the ANONYMOUS statement in FTP.DATA; however, the server is not allowing anonymous logins due to an FTP configuration error. The reason code indicates the nature of the configuration error. Furthermore, the FTP trace will contain messages further describing the error. When reason code is:

1
The FTP daemon was unable to locate the anonymous root directory. A possible reason for this is the userid specified on the ANONYMOUS statement is not defined in the user database.

2
The FTP daemon was unable to locate the executable file ls in the bin subdirectory of the anonymous root directory. A possible reason for this is the system programmer did not set up the root directory as directed in the z/OS Communications Server: IP User's Guide and Commands.

3
The FTP daemon was unable to locate the executable file sh in the bin subdirectory of the anonymous root directory. A possible reason for this is the system programmer did not set up the root directory as directed in the z/OS Communications Server: IP User's Guide and Commands.

4
The FTP daemon was unable to locate the executable file ftpdns in the /usr/sbin/ subdirectory of the anonymous root directory. A possible reason for this is the system programmer did not set up the root directory as directed in the z/OS Communications Server: IP User's Guide and Commands.

5
ANONYMOUSLEVEL value must be greater than or equal to 3 to support the SURROGATE parameter as an anonymous password.

System action

FTP logs the reason it is not allowing anonymous logins. FTP continues processing. Attempts to log in as userid anonymous will fail.

Operator response

Report the error to the system programmer.

System programmer response

If you do not want the FTP server to accept anonymous logins, remove the ANONYMOUS statement from FTP.DATA. If you do want the FTP server to accept anonymous logins, correct the error indicated by reason code. Inspecting the FTP trace for error messages related to not allowing anonymous logins might provide useful information. Trace must be active at startup to capture error messages related to anonymous FTP configuration. See z/OS Communications Server: IP User's Guide and Commands for information about configuring the FTP server for anonymous logins.

Module

EZAFTPDM

EZYFT74I Using internal translate tables for the control connection.

Explanation

A CTRLCONN statement in the FTP.DATA file specified a value of FTP_STANDARD_TABLE. FTP will use its internal translate tables, which are the same as the tables that are shipped in TCPXLBIN(STANDARD), for the control connection.

System action

Processing continues.
Using internal translate tables for the data connection.

Explanation
An SBDATACONN statement in the FTP.DATA file specified a value of FTP_STANDARD_TABLE. FTP will use its internal translate tables, which are the same as the tables that are shipped in TCPXLBIN(STANDARD), for the data connection.

System action
Processing continues.

EXTensions value value not recognized.

Explanation
While processing the FTP.DATA file, the server encountered the EXTensions statement with a value that was not SIZE, MDTM, REST_STREAM, AUTH_GSSAPI, AUTH_TLS, or UTF8. Only these values are valid for the EXTensions parameter.

value is the value encountered.

System action
The line containing the error is ignored. Processing of the FTP.DATA file continues.

Operator response
Contact the system programmer with the error message to correct the FTP.DATA file.

System programmer response
Correct the FTP.DATA file to contain a valid value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for information about the statements of the FTP.DAT file.
**Procedure name**
read_ftpData

**EZYFT77W** Unable to set address space nonswappable - *error*

**Explanation**
The FTP daemon was unable to set its address space to nonswappable as requested in the FTP.DATA file. *error* is the error message returned by the C run-time library for the failing __mlockall() routine.

**System action**
Processing continues. The FTP daemon will run with swappable memory.

**Operator response**
None.

**System programmer response**
Determine the meaning of the error message using z/OS Language Environment Runtime Messages. If the text of the message is EDC5139I Operation not permitted then verify that the daemon (FTPD) has at least READ access to the FACILITY class resource BPX.STOR.SWAP. See z/OS Communications Server: IP Configuration Reference for information about configuring the FTP daemon to be nonswappable.

**Module**
EZAFTPDM

**Procedure name**
set_nonswap

**EZYFT78I** lowport value *lowport* cannot exceed highport value *highport*

**Explanation**
FTP encountered an error while processing a PASSIVEDATAPORTS statement in the FTP.DATA file. The value specified for the lowest allowed port number is greater than the value specified for the highest allowed port number. This is not allowed.

*lowport* is the value specified for the lowest allowed port number.

*highport* is the value specified for the highest allowed port number.

**System action**
The current statement is ignored. FTP continues processing.

**Operator response**
Correct the erroneous statement in the FTP.DATA file, and restart FTP. See the z/OS Communications Server: IP Configuration Reference for information about statements in the FTP.DATA file.

**System programmer response**
None.

**Module**
EZAFTPEP
EZYFT80I   MODIFY COMMAND IGNORED - NO VALUE WAS SPECIFIED FOR trace

Explanation
A MODIFY command was issued to the FTP server to change the values for the FTP general trace or for the FTP extended trace. No trace value was specified. A command to change a trace must have at least one value specified.

trace is the trace that is being changed and can be one of the following:

- **DEBUG**  
  FTP general trace

- **DUMP**  
  FTP extended trace

System action
The command is ignored.

Operator response
Verify that the MODIFY command was entered correctly. See the z/OS Communications Server: IP Diagnosis Guide for information about diagnosing FTP server problems with traces and for the values of the DEBUG and DUMP parameters of the MODIFY command.

System programmer response
None.

Module
EZAFTPDF

Procedure name
mvs_command_handler

EZYFT81I   UNSUPPORTED trace VALUES - value_1 ... value_n

Explanation
A MODIFY command was issued to the FTP server to change the values for the FTP general trace or for the FTP extended trace.

trace is the trace that is being changed and can be one of the following:

- **DEBUG**  
  FTP general trace

- **DUMP**  
  FTP extended trace

The list shows the values that were specified that are not supported.

System action
The unsupported values are ignored.
Operator response

Verify that the MODIFY command was entered correctly. See the z/OS Communications Server: IP Diagnosis Guide for information about diagnosing FTP server problems with traces and for the values of the DUMP parameter of the MODIFY command that apply to JES.

System programmer response

None.

Module

EZAFTPDF

Procedure name

mvs_command_handler

EZYFT82I    ACTIVE SERVER TRACES - name_1 ... name_n

Explanation

This message displays the active FTP server general traces. This message is issued when a MODIFY command is issued to the FTP server to change the values for the FTP general trace.

name_x can be one of the following:

ACC
access control (logging in)

CMD
command trace

INT
program initialization and termination

FLO
function flow

FSC(1)
file services -- level 1

FSC(2)
file services -- level 2

FSC(3)
file services -- level 3

FSC(4)
file services -- level 4

FSC(5)
file services -- level 5

FSC(6)
file services -- level 6

FSC(7)
file services -- level 7

FSC(8)
file services -- level 8

JES
JES processing

NONE
no trace is active
PAR
  parser details
SEC
  security functions
SOC(1)
  socket services -- level 1
SOC(2)
  socket services -- level 2
SOC(3)
  socket services -- level 3
SOC(4)
  socket services -- level 4
SOC(5)
  socket services -- level 5
SOC(6)
  socket services -- level 6
SOC(7)
  socket services -- level 7
SOC(8)
  socket services -- level 8
SQL
  SQL processing
UTL
  utility functions

System action
FTP continues.

Operator response
None.

System programmer response
None.

Module
EZAFTPDF

Procedure name
mvs_command_handler

EZYFT83I  ACTIVE SERVER DUMPIDS - id_1 ... id_n

Explanation
A MODIFY command was issued to the FTP server to change the values for the FTP extended trace. This message displays the active FTP server extended trace IDs (DUMPIDS). If no extended trace IDs are active, then the word NONE is displayed.

System action
FTP continues.
Operator response
None.

System programmer response
None.

Module
EZAFTPDF

Procedure name
mvs_command_handler

EZYFT84I  parameter OBSOLETE - USE DUMP TO CONTROL FTP EXTENDED TRACING FOR JES

Explanation
A MODIFY command was issued to the FTP server to enable or disable the FTP extended trace for JES processing.

parameter is one of the following:

- JDUMP — enable extended tracing for JES
- NOJDUMP — disable extended tracing for JES

System action
The MODIFY command is ignored.

Operator response
See the z/OS Communications Server: IP Diagnosis Guide for information about diagnosing FTP server problems with Traces and for the values of the DUMP parameter of the MODIFY command that apply to JES.

System programmer response
None.

Module
EZAFTPDF

Procedure name
mvs_command_handler

EZYFT85I  PASSIVEDATACONN value must be UNRESTRICTED or NOREDIRECT

Explanation
While processing the FTP.DATA file, the server encountered the PASSIVEDATACONN parameter with a parameter value that was not UNRESTRICTED or NOREDIRECT. The only valid values for the PASSIVEDATACONN parameter are NORESTRICTED and NOREDIRECT.

System action
The line containing the error is ignored. The default of UNRESTRICTED will be used. Processing of the FTP.DATA file continues with the next line in the file.
Operator response
Contact the system programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the specified parameter. For more information, see z/OS Communications Server: IP Configuration Reference.

Module
EZAFTPEP

Procedure name
read_ftpdata

**EZYFT86I** USER TRACE IS OBSOLETE - parameter IS IGNORED

Explanation
A MODIFY command was issued to the FTP server to activate the user trace. The user trace is obsolete and is replaced by the user ID filter for the FTP server general and extended traces.

parameter is one of the following:

**UTRACE**
Start user trace

**NOUTRACE**
Stop user trace

System action
The MODIFY command is ignored.

Operator response
See the z/OS Communications Server: IP Diagnosis Guide for information about diagnosing FTP Server problems with traces and for the syntax of the user ID filter for the FTP server general and extended traces.

System programmer response
None.

Module
EZAFTPDF

Procedure name
mvs_command_handler

**EZYFT87I** VALUE NOT SUPPORTED FOR trace STATEMENT

Explanation
A trace statement in FTP.DATA specified a value that is not supported by FTP.

trace is one of the following:

**DEBUG**
FTP general trace
DUMP
FTP extended trace

System action
The statement is ignored.

Operator response
See the z/OS Communications Server: IP Configuration Reference for more information about the DEBUG and DUMP statements and FTP.DATA data set statements. See the z/OS Communications Server: IP User's Guide and Commands for information about changing local site defaults using FTP.DATA for the FTP client.

System programmer response
None.

Module
EZAFTPEP

Procedure name
read_ftpdata

EZYFT89I USERID FILTER - filter

Explanation
A MODIFY command was issued to the FTP server to change values for the FTP general trace (DEBUG) or extended trace (DUMP).

filter is the name of the filter for the userids for which tracing is active.

System action
None.

Operator response
None.

System programmer response
None.

Module
EZAFTPDF

Procedure name
mvs_command_handler

EZYFT90I IPADDR FILTER - filter

Explanation
A MODIFY command was issued to the FTP server to change values for the FTP general trace (DEBUG) or extended trace (DUMP).

filter is the name of the filter for the IP addresses for which tracing is active.
**System action**
None.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAFTPDF

**Procedure name**
mvs_command_handler

**EZYFT91I** PORTCOMMAND value must be ACCEPT or REJECT

**Explanation**
While processing the FTP.DATA file, the server encountered the PORTCOMMAND parameter with a parameter value that was not ACCEPT or REJECT. The only valid values for the PORTCOMMAND parameter are ACCEPT and REJECT.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

**Operator response**
Contact the system programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for more information about the PORTCOMMAND parameter.

**Module**
EZAFTPEP

**Procedure name**
read_ftpdata

**EZYFT92I** PORTCOMMANDPORT value must be UNRESTRICTED or NOLOWPORTS

**Explanation**
While processing the FTP.DATA file, the server encountered the PORTCOMMANDPORT parameter with a parameter value that was not UNRESTRICTED or NOLOWPORTS. The only valid values for the PORTCOMMANDPORT parameter are UNRESTRICTED and NOLOWPORTS.

**System action**
The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.
Operator response

Contact the system programmer with the error message to have the FTP.DATA file corrected.

System programmer response

Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for more information about the PORTCOMMANDIPADDR parameter.

Module

EZAFTPEP

Procedure name

read_ftpdata

EZYFT93I  PORTCOMMANDIPADDR value must be UNRESTRICTED or NOREDIRECT

Explanation

While processing the FTP.DATA file, the server encountered the PORTCOMMANDIPADDR parameter with a parameter value that was not UNRESTRICTED or NOREDIRECT. The only valid values for the PORTCOMMANDIPADDR parameter are NORESTRICTED and NOREDIRECT.

System action

The line containing the error is ignored. Processing of the FTP.DATA file continues with the next line in the file.

Operator response

Contact the system programmer with the error message to have the FTP.DATA file corrected.

System programmer response

Correct the FTP.DATA file to contain the correct value for the specified parameter. See the z/OS Communications Server: IP Configuration Reference for more information about the PORTCOMMANDIPADDR parameter.

Module

EZAFTPEP

Procedure name

read_ftpdata

EZYFT94I  ANONYMOUSLEVEL value must be greater than or equal to 3 to support the SURROGATE parameter as an anonymous password

Explanation

In the server FTP.DATA, SURROGATE is defined as the password for anonymous users, and ANONYMOUSLEVEL is less than 3. In order to use SURROGATE as the password for anonymous users, ANONYMOUSLEVEL must be greater than or equal to 3.

System action

The line containing the error is ignored. Anonymous logins will be disabled. FTP continues.

Operator response

Contact the system programmer with the error message to have the FTP.DATA file corrected.
System programmer response
Set the ANONYMOUSLEVEL to 3 or greater. See the z/OS Communications Server: IP Configuration Reference for more information about ANONYMOUSLEVEL.

Module
EZAFTPDM

Procedure name
check_surrogate_support

EZYFT95I  Server setup for TLS failed

Explanation
The FTP server detected an error condition when it was processing a request for TLS authentication from an FTP client. The error condition prevented the server from completing the request for a TLS protected session.

System action
The request for authentication failed. If a secure FTP session is required by the server, the session is active but no commands are accepted until an AUTH command is successfully processed. If a secure FTP session is not required, the session setup continues but the session is not TLS protected.

Operator response
Contact the system programmer.

System programmer response
Determine the cause of the error by examining the FTP server trace. The SEC trace type should be active before the client attempts to connect to the FTP server. The trace entry that precedes message EZYFT95I in the trace describes the reason for the failure.

The error message is displayed when one of the following has occurred:

- No keyring is defined in the FTP.DATA file for the server or a keyring file name is specified but there is no associated stash file. See z/OS Communications Server: IP Configuration Reference for more information about KEYRING.
- One of the TLS setup functions used by the FTP server failed. See z/OS Cryptographic Services System SSL Programming to determine how to obtain diagnostic data for the TLS programming functions.

Module
EZAFTPFR, EZAFTPRX

Procedure name
auth, main

EZYFT96I  TLS handshake failed

Explanation
The FTP server detected an error condition when it was processing a request for TLS authentication from an FTP client. The error condition occurred during the server to client handshake process and prevented the server from completing the request for a TLS protected session.
**System action**
The request for authentication failed. If a secure FTP session is required by the server, the session is active but no commands are accepted until an AUTH command is successfully processed. If a secure session is not required, the session setup continues but the session is not TLS protected.

**Operator response**
Contact the system programmer.

**System programmer response**
Determine the cause of the error by examining the FTP server trace. The SEC trace type and the SOC level 3 trace type must be active before the client attempts to connect to the FTP server. The trace entries that precede this message in this trace describe the reason for the failure.

**Module**
EZAFTPFR, EZAFTPRX

**Procedure name**
auth, main

**EZYFT97I**  SECURE_FTP value must be REQUIRED or ALLOWED

**Explanation**
While processing the FTP.DATA file, the server or client encountered the SECURE_FTP statement with a parameter value that was not REQUIRED or ALLOWED. The only valid values are REQUIRED and ALLOWED.

**System action**
The line containing the error is ignored. FTP.DATA file processing continues with the next line of the file.

**Operator response**
Contact the system programmer with the error message to have the FTP.DATA file corrected.

**System programmer response**
Correct the FTP.DATA file to contain the correct value for the statement. See the z/OS Communications Server: IP Configuration Guide for information about configuring the FTP server to use the TLS security mechanism.

**Module**
EZAFTPEP

**Procedure name**
read_ftpdata

**EZYFT98I**  SECURE_LOGIN value must be NO_CLIENT_AUTH, REQUIRED, or VERIFY_USER

**Explanation**
While processing the FTP.DATA file, the server encountered the SECURE_LOGIN statement with a parameter value that was not NO_CLIENT_AUTH, REQUIRED, or VERIFY_USER. The only valid values are NO_CLIENT_AUTH, REQUIRED, and VERIFY_USER.
System action
The line containing the error is ignored. FTP.DATA file processing continues with the next line of the file.

Operator response
Contact the system programmer with the error message to have the FTP.DATA file corrected.

System programmer response
Correct the FTP.DATA file to contain the correct value for the statement. See the z/OS Communications Server: IP Configuration Guide for information about configuring the FTP server to use the TLS security mechanism.

Module
EZAFTPEP

Procedure name
read_ftpdata

EZYFT99I Domain name unknown. Getaddrinfo() rc getaddrinfo_rc.

Explanation
The FTP server was not able to determine its domain name because the getaddrinfo() resolver call failed.

getaddrinfo_rc is the return code from getaddrinfo(). See the z/OS XL C/C++ Runtime Library Reference for information about getaddrinfo() return codes.

System action
FTP continues processing. The host domain name will not be known to the server.

Operator response
Contact the system programmer with the error.

System programmer response
Ensure that the host name is accessible through a name server, or defined in a local hosts or ipnodes data set. See the z/OS Communications Server: IP Configuration Guide for information about domain name systems, local hosts data sets, and ipnodes data sets.

Module
ezaftpsd

Procedure name
handle_client_socket()
Chapter 10. EZYPxxxx messages

**EZYPR001**  There were no records to satisfy the request

**Explanation**
No records matching the selection criteria were found.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPQ

**EZYPR002**  Invalid Trace Option, please enter a 1 or 2

**Explanation**
The value specified in the Trace Option field is not valid.

**System action**
The system continues processing.

**Operator response**
Specify a valid value in the Trace Option field and reissue the request. A valid trace option is a "1" or "2".

**System programmer response**
None.

**Module**
EZAPPDGP

**EZYPR003**  Invalid Confirm option, please enter a "/" or clear field

**Explanation**
The value specified in the Confirm option field is not valid.

**System action**
The system continues processing.

**Operator response**
Specify a valid value in the Confirm option field and reissue the request. A valid confirm option is a "/" or blank.
System programmer response
None.

Module
EZAPPDCP

EZYPR004  Dataset name must be in valid dataset format

Explanation
The data entered for the data set name contains an invalid character.

System action
The system continues processing.

Operator response
Enter valid data for the data set name field. The data set name must follow the standard TSO data set naming rules. See z/OS TSO/E User's Guide for more information.

System programmer response
None.

Module
various

EZYPR005  Delete request for the record was canceled

Explanation
A delete request was issued for a selected record, but it was canceled by the user. As a result, the record is not deleted.

System action
The system continues processing.

Operator response
None.

System programmer response
None.

Module
various

EZYPR006  Queue record requested not found

Explanation
The requested queue file record cannot be processed because it does not exist.

System action
The system continues processing.
**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPQ

**EZYPR007** Table has mismatched record(s); record refresh was issued.

**Explanation**
For one or more queue file records, a requested operation was not performed because of a mismatch between the data shown on the NPF Queue List panel and the data currently in the record.

**System action**
For each mismatched record, the requested operation is not performed; the data on the NPF Queue List panel is updated to match the current record data, and a '?' is displayed in the Action column.

**Operator response**
For each mismatched record, check the updated data on the Queue List panel to determine if the requested operation is still needed. If so, repeat the request.

**System programmer response**
None.

**Module**
EZAPPQ

**EZYPR008** Incorrect call to EZAPPQ

**Explanation**
An invalid call to module EZAPPQ from panel EZAPPMP was issued. Expected parameters in the call might have been missing.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Contact the IBM Software Support Center.

**Module**
EZAPPQ

**EZYPR009** Cannot create queue table
**Explanation**
An error occurred during the creation of a queue table. The queue table is used for displaying the list of queue records.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Contact the IBM Software Support Center.

**Module**
EZAPPQ

**EZYPR010 Queue file open failed**

**Explanation**
Either the queue file cannot be opened, access to the queue file is denied, or the queue file does not exist.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Verify the existence of the queue file. If it exists, then determine if the queue file is read-protected. If it is, consult your Resource Access Control Facility (RACF) administrator. If the queue file is not read-protected and the error still occurs, contact the IBM Software Support Center.

**Module**
EZAPPQ

**EZYPR011 Queue file read failed**

**Explanation**
An error occurred during the attempt to read a queue file record from the queue file.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Contact the IBM Software Support Center.
EZAPPQ

**EZYPR014  Queue file update failed**

**Explanation**
A requested update to a queue file record could not be completed because of an I/O error.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Determine the cause of the I/O error and take appropriate action.

---

EZAPPQ

**EZYPR015  Invalid return code from ISPF**

**Explanation**
The ISPF dialog management services detected an error during panel processing.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Contact the IBM Software Support Center.

---

various

**EZYPR016  Invalid command entered**

**Explanation**
The data entered in the command line area is not valid.

**System action**
The system continues processing.

**Operator response**
None.
System programmer response
Correct the command and reissue the request.

Module
various

**EZYPR017**  Trace file cannot be opened

**Explanation**
Either the trace file cannot be opened, the trace file cannot be dynamically allocated, access to the trace file is denied, or the trace file does not exist.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Verify the type of data set prefix used. If the prefix is a TSO user ID, the trace file should be dynamically allocated during the open attempt. If the dynamic allocation failed, contact the IBM Software Support Center. If the prefix is other than a TSO user ID, verify the existence of the trace file. If it exists, then determine if the trace file is write-protected. If it is, consult your Resource Access Control Facility (RACF) administrator. If the trace file is not write-protected and the error still occurs, contact the IBM Software Support Center.

Module
EZAPPDG

**EZYPR018**  You do not have access authorization for this dataset

**Explanation**
The data set is security protected for read or update by the Resource Access Control Facility (RACF). All access attempts to this data set will be denied.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Contact your RACF administrator if you want access permission to this data set.

Module
various

**EZYPR019**  You do not have update authorization for this dataset
Explanation
The data set is security protected for update by the Resource Access Control Facility (RACF). The data set is available for read-only. You cannot edit a record in the data set, but you can browse a record in the data set.

System action
The system continues processing.

Operator response
None.

System programmer response
Contact your RACF administrator if you want update permission to this data set.

Module
various

EZYPR020 Last Send Time is not valid

Explanation
The value specified in the Last Send Time field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid Last Send Time and reissue the request.

Module
EZAPPQRP

EZYPR021 Last Send Date is not valid

Explanation
The value specified in the Last Send Date field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid Last Send Date and reissue the request.
Module
EZAPPQRP

**EZYPR022 Printer Name must be specified**

**Explanation**
The value specified in the Printer Name field is not valid.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify a valid Printer Name and reissue the request.

Module
various

**EZYPR023 Host Name or IP address must be specified**

**Explanation**
The value specified in the Host Name or IP address field is not valid.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify a valid Host Name or IP address and reissue the request.

Module
various

**EZYPR024 Command is not valid for the current record status**

**Explanation**
An operation requested for a queue record was not performed because the request is not allowed for records with this data set status value.

**System action**
The system continues processing.

**Operator response**
Use a command that is valid for the current status of the queue record you are working with.
**System programmer response**
None.

**Module**
EZAPPQ.

**EZYPR025  Record is currently in use; browse or retry later**

**Explanation**
A requested operation cannot be performed at this time for a specified queue record file because the record is being updated by either the NPF Queue Manager or another panel operator.

**System action**
The system continues processing.

**Operator response**
Browse the queue file record if read-only access is sufficient; otherwise, try the original operation again later.

**System programmer response**
None.

**Module**
EZAPPQ.

**EZYPR026  Queue record was changed; please re-enter your changes.**

**Explanation**
An edit operation on a queue file record took too long (over 12 minutes) allowing the NPF Queue Manager or another panel operator to access and modify that record. All updates from the attempted edit operation have been lost and must be re-entered.

**System action**
The system continues processing.

**Operator response**
Repeat the edit operation, being careful to complete the operation within 12 minutes which is the maximum time for which you are guaranteed exclusive access to the record.

**System programmer response**
None.

**Module**
EZAPPQ.

**EZYPR027  Queue record with status X can only be browsed**

**Explanation**
An edit request has been rejected for a queue record whose data set status = ‘X’.

**Module**
EZAPPQ.
System action
The system continues processing.

Operator response
Do not try to edit queue records whose data set status = ‘X’. Browse is the only valid operation for this type of record.

System programmer response
None.

Module
EZAPPQ.

EZYPR028  Queue record with status X cannot be deleted

Explanation
A delete request has been rejected for a queue record whose data set status = ‘X’.

System action
The system continues processing.

Operator response
Do not try to delete queue records whose data set status = ‘X’. Browse is the only valid operation on this type of record.

System programmer response
None.

Module
EZAPPQ

EZYPR030  Dataset type is in use by another dialog on this userid

Explanation
An attempt to process an NPF routing, options or queue file was rejected because another dialog under the same userid was already processing a file of the same type.

System action
The system continues processing.

Operator response
When doing NPF processing from multiple dialogs under a single userid, have each dialog process a different type of file. For example, it is acceptable to process a routing file from one dialog and an options file from another dialog.

System programmer response
None.
Module
EZAPPQ or EZAPPC

EZYPR031  Options file record already exists.

Explanation
The specified new options record cannot be added to the options file because that record already exists.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a new options name for the key field other than the one that already exists and reissue the request.

Module
EZAPPC

EZYPR032  Options file record is not added.

Explanation
The options file record was not added, or you do not have write access to the options file.

System action
The system continues processing.

Operator response
None.

System programmer response
Check the options file for correct information and reissue the request. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPC

EZYPR033  Options file record is added.

Explanation
The new record was added to the options file as requested.

System action
The system continues processing.

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

**Module**
EZAPPC

**EZYPR034**  Options file record does not exist.

**Explanation**
The system cannot delete the options file record you requested because it does not exist.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Select an existing options file record that you want deleted and reissue the request.

**Module**
EZAPPC

**EZYPR035**  Options file record is not deleted.

**Explanation**
The options file record cannot be deleted.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Check the options file for correct information and reissue the request. If the error still occurs, contact the IBM Software Support Center.

**Module**
EZAPPC

**EZYPR036**  Options file record is deleted.

**Explanation**
The options file record was deleted as requested.

**System action**
The system continues processing.
Operator response
None.

System programmer response
None.

Module
EZAPPPC

EZYPR037 Type of option selected is not valid.

Explanation
The option selected is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Select a valid option and enter the request.

Module
various

EZYPR038 Missing broadcast rec(s) detected; Issue "CANcel" or complete

Explanation
More than one destination was specified for the normal route, and data has not been entered for the specific routing broadcasts.

System action
The system continues processing.

Operator response
None.

System programmer response
Enter data for the specific routing broadcasts on the routing broadcast record panel. If a normal route and no specific routing broadcasts are wanted, then change the value in the number of destinations field to "1". If the data for normal and specific routing records are to be discarded, issue the cancel command on the command line.

Module
EZAPPPC

EZYPR041 Old options file record does not exist.
Explanation
The copy request cannot be completed because the old options file record specified does not exist.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify an old options file record that exists and reissue the request.

Module
EZAPPPC

EZYPR042  New options file record is not copied.

Explanation
The new options file record was not copied.

System action
The system continues processing.

Operator response
None.

System programmer response
Check the options file for correct information and reissue the request. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

EZYPR043  New options file record is copied.

Explanation
The new options file record was copied as requested.

System action
The system continues processing.

Operator response
None.

System programmer response
None.
Module
EZAPPPC

**EZYPR044**   Options file record does not exist.

**Explanation**
The requested options file record cannot be updated because it does not exist.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify an options file record that exists and reissue the request.

Module
EZAPPPC

**EZYPR045**   Options file record is not updated.

**Explanation**
The options file record was not updated.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Check the options file for correct information and reissue the request. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

**EZYPR046**   Options file record is updated.

**Explanation**
The options file record was updated as requested.

**System action**
The system continues processing.

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

Module
EZAPPPC

EZYPR048  New options file record already exists.

Explanation
The options file record cannot be copied because a new options file record already exists.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify an options file record other than the one that already exists and reissue the request.

Module
EZAPPPC

EZYPR049  Select an option.

Explanation
An option must be selected from this panel.

System action
The system continues processing.

Operator response
None.

System programmer response
Select an option from the panel and issue the request.

Module
various

EZYPR051  Enter options name for the key-field.

Explanation
The options name for the key-field information is missing from the request.

System action
The system continues processing.
Operator response
None.

System programmer response
Enter a value in the options name for the key-field and reissue the request.

Module
EZAPPN1

EZYPR052  Input record exit is not valid.

Explanation
The value specified in the input record exit field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid input record exit value and reissue the request.

Module
EZAPPN7

EZYPR053  Old options name is not valid.

Explanation
The value specified in the old options name field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid value in the old options name field and reissue the request.

Module
EZAPPN4

EZYPR054  New options name is not valid.

Explanation
The value specified in the new options name field is not valid.
System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid value in the new options name field and reissue the request.

Module
EZAPPN4

EZYPR055  Tracing is enabled

Explanation
The request to enable NPF ISPF tracing is acknowledged, and the specified trace data set is opened to receive trace information.

System action
The system continues processing.

Operator response
None.

System programmer response
None.

Module
EZAPPDG

EZYPR056  Tracing is disabled

Explanation
The request to disable NPF ISPF tracing is acknowledged, and the specified trace data set is closed.

System action
The system continues processing.

Operator response
None.

System programmer response
None.

Module
EZAPPDG

EZYPR057  Old options name must be specified.
**Explanation**  
The old options name for the key-field is missing.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Enter a value for the old options name key-field and reissue the request.

**Module**  
EZAPPN4

**EZYPR058   New options name must be specified.**

**Explanation**  
The new options name for the key-field is missing.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Specify a value for the new options key-field and reissue the request.

**Module**  
EZAPPN4

**EZYPR060   Press ENTER to save data before proceeding to next record**

**Explanation**  
During add processing for a multiple-destination routing, the DOWN PF key was pressed to proceed to the next destination record before the ENTER key was pressed to save the data for the current destination record.

**System action**  
The DOWN PF key is ignored and the panel continues to display the current destination record.

**Operator response**  
Press the ENTER key to save the data for the current destination record before attempting to process the next record.

**System programmer response**  
None.
Module
EZAPPPC

**EZYPR061**  Normal route record already exists.

**Explanation**
The specified new route cannot be added to the routing file because that route already exists.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify a normal route other than the one that already exists and reissue the request.

Module
EZAPPPC

**EZYPR062**  Normal route record is not added.

**Explanation**
The normal route was not added.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Try the request again. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

**EZYPR063**  Normal route record is added.

**Explanation**
A normal route was added to the routing file.

**System action**
The system continues processing.

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

Module
EZAPPPC

**EZYPR064**  Normal route record does not exist.

Explanation
The route specified in the delete request does not exist, so the request cannot be completed.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a route that exists and reissue the request.

Module
EZAPPPC

**EZYPR065**  Normal route record is not deleted.

Explanation
The normal route was not deleted, or you do not have write access to the routing file.

System action
The system continues processing.

Operator response
None.

System programmer response
Try the request again. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

**EZYPR066**  Normal route record is deleted.

Explanation
The normal route is deleted from the routing file as requested.

System action
The system continues processing.
Operator response
None.

System programmer response
None.

Module
EZAPPPC

EZYPR068  Routing file cannot be opened.

Explanation
Either the routing file cannot be opened, access to the routing file is denied, or the routing file does not exist.

System action
The system continues processing.

Operator response
None.

System programmer response
Verify the existence of the routing file. If it exists, then determine if the routing file is read-protected. If it is, consult your Resource Access Control Facility (RACF) administrator. If the routing file is not read-protected and the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

EZYPR069  Routing file cannot be closed.

Explanation
The system detected an error. The routing file cannot be closed.

System action
The system continues processing.

Operator response
None.

System programmer response
Try the request again. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

EZYPR071  Old normal route record does not exist.

Explanation
The normal route record cannot be copied because the old normal route does not exist.
**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify major and minor names for an old normal route record that exists and reissue the request.

**Module**
EZAPPPC

---

**EZYPR072 Normal route record is not copied**

**Explanation**
The normal route record was not copied.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Reissue the request. If the error still occurs, contact the IBM Software Support Center.

**Module**
EZAPPPC

---

**EZYPR073 Normal route record is copied**

**Explanation**
The request to copy the normal route record is complete.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPPC

---

**EZYPR074 Specific broadcast record does not exist.**
**Explanation**  
A specific broadcast record cannot be updated because it does not exist.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Because the specific broadcast record does not exist, all of the routing records associated with this missing broadcast record are not valid. Issue the delete request to delete the obsolete records and then issue the add request to create the new routing records.

**Module**  
EZAPPPC

**EZYPR075 Normal route record is not updated**

**Explanation**  
The normal route record was not updated.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Reissue the request. If the error still occurs, contact the IBM Software Support Center.

**Module**  
EZAPPPC

**EZYPR076 Normal route record is updated.**

**Explanation**  
The normal route record was updated as requested.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
None.
Module
EZAPPPC

EZYPR078  New normal route record already exists.

Explanation
The normal route record cannot be copied because a new normal route record already exists.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify major and minor names for a routing file record other than the one that already exists and reissue the request.

Module
EZAPPPC

EZYPR079  Old specific broadcast record does not exist.

Explanation
The old specific broadcast record associated with the normal routing record in the request does not exist.

System action
The system continues processing.

Operator response
None.

System programmer response
Because the old specific broadcast record does not exist, all of the routing records associated with this missing broadcast record are not valid. Issue the delete request to delete the obsolete records. Either specify major and minor names for a different routing record and reissue the request again or issue an add request to create new routing records which can be used for the copy request again.

Module
EZAPPPC

EZYPR081  Enter old major name for the key-field.

Explanation
The request is missing an old major name.

System action
The system continues processing.
Operator response
None.

System programmer response
Specify an old major name and reissue the request. A valid major name is an 8-byte alphanumeric name.

Module
EZAPPN12

EZYPR082  Enter new major name for the key-field.

Explanation
The request is missing a new major name.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a new major name and reissue the request. A valid major name is an 8-byte alphanumeric name.

Module
EZAPPN12

EZYPR083  Enter old minor name for the key-field.

Explanation
The request is missing an old minor name for the key-field.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify an old minor name and reissue the request. A valid minor name is an 8-byte alphanumeric name.

Module
EZAPPN12

EZYPR084  Enter new minor name for the key-field.

Explanation
The request is missing a new minor name for the key-field.
**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Specify a new minor name and reissue the request. A valid minor name is an 8-byte alphanumeric name.

**Module**  
EZAPPN12

**EZYPR086  Enter major name for the key-field**

**Explanation**  
The major name is missing from the request.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Specify a major name and reissue the request. A valid major name is an 8-byte alphanumeric name.

**Module**  
various

**EZYPR087  Enter minor name for the key-field.**

**Explanation**  
The minor name is missing from the request.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Specify a minor name and reissue the request. A valid minor name is an 8-byte alphanumeric name.

**Module**  
various

**EZYPR089  Number of destination(s) must be specified.**
Explanation
The destination number is missing from the request.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a destination value and reissue the request. A valid value is in the range 1 to 65535.

Module
EZAPPN15

EZYPR091 Retain time (successful) must be specified

Explanation
The retain time (successful) is missing from the request.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify the retain time (successful) and reissue the request. Retain time must be in the format dddhhmm with values as follows:

\[
\begin{align*}
\text{ddd} & : 000 \text{ to } 366 \\
\text{hh} & : 00 \text{ to } 23 \\
\text{mm} & : 00 \text{ to } 59
\end{align*}
\]

Module
EZAPPN15

EZYPR092 Retain time (unsuccessful) must be specified.

Explanation
The retain time (unsuccessful) is missing from the request.

System action
The system continues processing.
**System programmer response**
Specify the retain time (unsuccessful) and reissue the request. Retain time must be in the format $dddhhmm$ with values as follows:

- **$ddd$**
  - 000 to 366

- **$hh$**
  - 00 to 23

- **$mm$**
  - 00 to 59

**Module**
EZAPPN15

**EZYPR093** Retry interval time must be specified.

**Explanation**
The retry interval time is missing from the request.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify the retry interval time and reissue the request. Retry interval time must be in the format $dddhhmm$ with values as follows:

- **$ddd$**
  - 000 to 366

- **$hh$**
  - 00 to 23

- **$mm$**
  - 00 to 59

**Module**
EZAPPN15

**EZYPR094** Retry limit must be specified.

**Explanation**
The retry limit is missing from the request.

**System action**
The system continues processing.
**Operator response**
None.

**System programmer response**
Specify a retry limit and reissue the request. A valid retry limit is in the range 0 to 65535.

**Module**
EZAPPN15

**EZYPR095  Options name must be specified.**

**Explanation**
The options name is missing from the request.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify an options name and reissue the request. A valid options name is a 16-character alphanumeric name.

**Module**
EZAPPN15

**EZYPR097  Specific broadcast record is not updated.**

**Explanation**
The system detected an error. The specific broadcast record update request was not completed.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Try the request again. If the error still occurs, contact the IBM Software Support Center.

**Module**
EZAPPPC

**EZYPR098  Specific broadcast record is updated.**

**Explanation**
The specific broadcast record update request completed.
**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
EZAPPPC

**EZYPR099**  
Specific broadcast record already exists.

**Explanation**  
The specific broadcast record cannot be added to the routing file because that route already exists.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Because the specific broadcast record already exists, the created normal routing record is not valid. Issue the delete request to delete the obsolete records. Issue an add request to create the new routing records again.

**Module**  
EZAPPPC

**EZYPR101**  
Specific broadcast record is added.

**Explanation**  
A specific broadcast route was added to the routing file.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
EZAPPPC

**EZYPR102**  
Specific broadcast is not added.
**Explanation**
The system detected an error. The specific broadcast was not added.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Restart the Network Print Facility and reissue the add request. If the error still occurs, contact the IBM Software Support Center.

**Module**
EZAPPC

**EZYPR104  Number of destination(s) is not valid**

**Explanation**
The number of destinations value must be larger than 0.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify a valid value for the number of destinations. A valid value is from 1 to 65535.

**Module**
various

**EZYPR105  Specific broadcast record is not deleted.**

**Explanation**
The system detected an error. The specific broadcast record delete request was not completed.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Try the request again. If the error still occurs, contact the IBM Software Support Center.
Module
EZAPPC

EZYPR106  Normal route & specific broadcast record(s) are deleted.

Explanation
The normal route and specific broadcast records delete request completed.

System action
The system continues processing.

Operator response
None.

System programmer response
None.

Module
EZAPPC

EZYPR107  Specific broadcast record is not copied.

Explanation
The system detected an error. The specific broadcast record copy request was not completed.

System action
The system continues processing.

Operator response
None.

System programmer response
Try the request again. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPC

EZYPR108  Normal route & specific broadcast record(s) are copied.

Explanation
The normal route and specific broadcast records copy request completed.

System action
The system continues processing.

Operator response
None.
System programmer response
None.

Module
EZAPPPC

**EZYPR111**  LU class is not valid.

Explanation
The LU class value entered is not valid. Valid values are 1 to 64.

System action
The system continues processing.

Operator response
Specify a valid LU class value and reissue the request.

System programmer response
None.

Module
various

**EZYPR112**  LU class contains invalid character.

Explanation
A character in the LU class field is not valid. Valid values are 1 to 64.

System action
The system continues processing.

Operator response
Specify a valid LU class value and reissue the request.

System programmer response
None.

Module
EZAPPPC

**EZYPR113**  Incorrect call to EZAPPPC

Explanation
The call to module EZAPPPC from panel EZAPPMP was not valid. Expected parameters in the call might have been missing.

System action
The system continues processing.
**Operator response**
None.

**System programmer response**
Contact the IBM Software Support Center.

**Module**
EZAPPPC

**EZYPR116** Normal route & specific broadcast record(s) are not added

**Explanation**
The normal and/or specific broadcast records were added during the add request, but a cancel command was issued to discard the added records.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
None.

**Module**
EZAPPPC

**EZYPR117** Normal route is copied but some data is lost.

**Explanation**
The normal route copy request completed, but not all of the data was copied.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Submit the copy request again. If the error still occurs, contact the IBM Software Support Center.

**Module**
EZAPPPC

**EZYPR131** Old major name is not valid.

**Explanation**
The value specified in the old major name field is not valid.
**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify a valid old major name and reissue the request. A valid old major name is an 8-byte alphanumeric name.

**Module**
EZAPPN12

**EZYPR132  Old minor name is not valid.**

**Explanation**
The value specified in the old minor name field is not valid.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify a valid old minor name and reissue the request. A valid old minor name is an 8-byte alphanumeric name.

**Module**
EZAPPN12

**EZYPR133  New major name is not valid.**

**Explanation**
The value specified in the new major name field is not valid.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify a valid new major name and reissue the request. A valid new major name is an 8-byte alphanumeric name.

**Module**
EZAPPN12

**EZYPR134  New minor name is not valid.**
Explanation
The value specified in the new minor name field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid new minor name and reissue the request. A valid new minor name is an 8-byte alphanumeric name.

Module
EZAPPN12

EZYPR135  Major name is not valid.

Explanation
The value specified in the major name field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid major name and reissue the request. A valid major name is an 8-byte alphanumeric name.

Module
various

EZYPR136  Minor name is not valid.

Explanation
The value specified in the minor name field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid minor name and reissue the request. A valid minor name is an 8-byte alphanumeric name.
Module various

EZYPR141 Options name is not valid.

Explanation
The value specified in the Options name field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid value in the Options name field and reissue the request. A valid options name is a 16-character alphanumeric name.

Module various

EZYPR142 Routing exit is not valid.

Explanation
The value specified in the Routing exit field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid value in the Routing exit field and reissue the request. A valid routing exit name is an 8-character alphanumeric name.

Module EZAPPN15

EZYPR143 Normal route & specific broadcast record(s) not deleted

Explanation
The system detected an error. The normal and specific broadcast records were not deleted as a result of the delete request.

System action
The system continues processing.
Operator response
None.

System programmer response
Try the request again. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

EZYPR144 Unable to copy old specific broadcast record.

Explanation
The system detected an error. A specific broadcast record associated with the normal routing record could not be copied.

System action
The system continues processing.

Operator response
None.

System programmer response
Because the old specific broadcast record is not valid, all of the copied routing records with this broadcast record are deleted. Issue the delete request to delete the obsolete records for the old routing records. Either specify major and minor names for a different routing record and reissue the request again, or issue an add request to create new routing records which can be used for the copy request again.

Module
EZAPPPC

EZYPR147 Unable to copy old normal route record.

Explanation
The system detected an error. A copy cannot be done for the old normal route record specified.

System action
The system continues processing.

Operator response
None.

System programmer response
Try the request again. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

EZYPR148 Normal route record is added but some data is lost.
Explanation
The normal route record add request completed, but not all of the data was added.

System action
The system continues processing.

Operator response
None.

System programmer response
Submit the add request again. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

EZYPR149 Specific broadcast record is added but some data is lost.

Explanation
The specific broadcast record add request completed, but not all of the data was added.

System action
The system continues processing.

Operator response
None.

System programmer response
Submit the add request again. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPPC

EZYPR161 Options file cannot be opened.

Explanation
Either the options file cannot be opened, access to the options file is denied, or the options file does not exist.

System action
The system continues processing.

Operator response
None.

System programmer response
Verify the existence of the options file. If it exists, then determine if the options file is read-protected. If it is, consult your Resource Access Control Facility (RACF) administrator. If the routing file is not read-protected and the error still occurs, contact the IBM Software Support Center.
Module
EZAPPC

**EZYPR162 Options file cannot be closed.**

**Explanation**
The system detected an error. The options file specified cannot be closed.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Try the request again. If the error still occurs, contact the IBM Software Support Center.

Module
EZAPPRT

**EZYPR180 SLU/Dest is not valid**

**Explanation**
The value specified in the SLU/Dest field is not valid.

**System action**
The system continues processing.

**Operator response**
None.

**System programmer response**
Specify a valid SLU/Dest value and reissue the request.

Module
EZAPPQSP

**EZYPR181 PLU/JOBNNAME is not valid**

**Explanation**
The value specified in the PLU/JOBNNAME field is not valid.

**System action**
The system continues processing.

**Operator response**
None.
System programmer response
Specify a valid PLU/JOBNAME value and reissue the request.

Module
EZAPPQSP

EZYPR182  Creation Date is not valid

Explanation
The value specified in the Creation Date field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid Creation Date and reissue the request.

Module
EZAPPQSP

EZYPR183  Creation Time is not valid

Explanation
The value specified in the Creation Time field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid Creation Time and reissue the request.

Module
EZAPPQSP

EZYPR184  Dataset Status is not valid

Explanation
The value specified in the Dataset Status field is not valid.

System action
The system continues processing.
Operator response
None.

System programmer response
Specify a valid Dataset Status and reissue the request.

Module
EZAPPQSP

EZYPR187  Invalid SEQ value - enter KEY or TIME

Explanation
The SEQ field on the NPF Queue Selection panel contains an invalid value.

System action
The system continues processing.

Operator response
Set the SEQ field to either KEY or TIME to indicate the order in which the selected queue records should be displayed on the Queue List panel.

System programmer response
None.

Module
EZAPPQ

EZYPR190  Next Send Time is not valid

Explanation
The value specified in the Next Send Time field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid Next Send Time value and reissue the request.

Module
EZAPPQSP

EZYPR191  Next Send Date is not valid

Explanation
The value specified in the Next Send Date field is not valid.
System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid Next Send Date value and reissue the request.

Module
EZAPPQSP

EZYPR192   Retry Interval is not valid

Explanation
The value specified in the Retry Interval field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid Retry Interval value and reissue the request.

Module
various

EZYPR193   Retry Attempts are not valid

Explanation
The value specified in the Retry Attempts field is not valid.

System action
The system continues processing.

Operator response
None.

System programmer response
Specify a valid Retry Attempts value and reissue the request.

Module
EZAPPQRP

EZYPR194   Retry Limit is not valid
**Explanation**  
The value specified in the Retry Limit field is not valid.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Specify a valid Retry Limit value and reissue the request.

**Module**  
various

**EZYPR195 Retain Time Successful is not valid**

**Explanation**  
The value specified in the Retain Time Successful field is not valid.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Specify a valid Retain Time Successful value and reissue the request.

**Module**  
various

**EZYPR196 Retain Time Unsuccessful is not valid**

**Explanation**  
The value specified in the Retain Time Unsuccessful field is not valid.

**System action**  
The system continues processing.

**Operator response**  
None.

**System programmer response**  
Specify a valid Retain Time Unsuccessful value and reissue the request.
**EZYPR213  Default page format name is not valid**

**Explanation**
The value specified in the Def Page Format field is not valid.

**System action**
The system continues processing.

**Operator response**
Either leave the field blank or enter the name of a valid Default Page Format table entry. See the z/OS Communications Server: IP Network Print Facility for more information.

**System programmer response**
None.

**Module**
EZAPPN15

---

**EZYPR214  End of file name is not valid.**

**Explanation**
The value entered for Eofile Name field is not valid.

**System action**
The system continues processing.

**Operator response**
Either leave the field blank or enter the name of a valid End-of-File Rules table entry. See the z/OS Communications Server: IP Network Print Facility for more information.

**System programmer response**
None.
Chapter 11. EZYRxxxx messages

EZYRC01I  Calling function rexec with the following:

Explanation
This is the first of a two part message. It is followed by EZYRC02I which lists the host, user, cmd, and port that are being passed to the rexec function that is provided by z/OS UNIX System Services and that does the work of passing the rexec information to the server.

System action
OREXEC continues.

Operator response
None.

System programmer response
None.

Module
rexec1.c

Procedure name
main()

EZYRC02I  Host: AAA user BBB, cmd CCC, port DDD

Explanation
This is the information that is being passed to the rexec function. AAA is the host that the string is being passed to, BBB is the logon ID of the user on the remote system, CCC is the command that is being passed to the remote system and DDD is the port number.

System action
OREXEC continues.

Operator response
None.

System programmer response
None.

Module
rexec1.c

Procedure name
main()

EZYRC03E  The call to rexec procedure failed.
**Explanation**
The call to the rexec procedure that will do the actual passing of the information to the remote host failed.

**System action**
OREXEC ends.

**Operator response**
By examining the command typed or using the -d option the problem should be apparent. If inetd is down on the remote host, this message will appear.

**System programmer response**
None.

**Module**
rexec1.c

**Procedure name**
main()

---

**EZYRC04I**
Usage: orexec -d -l user -p pwd

**Explanation**
This the first of a group of messages that is issued when a question mark has been entered, or an invalid number of arguments have been entered.

**System action**
OREXEC ends.

**Operator response**
Correct the invalid argument and try the command again.

**System programmer response**
None.

**Module**
rexec1.c

**Procedure name**
usage()

---

**EZYRC05I**
options: -

**Explanation**
See message EZYRC04I.
Operator response
By examining the command typed or using the -d option the problem should be apparent.

System programmer response
None.

Module
rexec1.c

Procedure name
usage()

EZYRC06I -? display this message

Explanation
See message EZYRC04I.

System action
OREXEC ends.

Operator response
By examining the command typed or using the -d option the problem should be apparent.

System programmer response
None.

Module
rexec1.c

Procedure name
usage()

EZYRC07I -d turn on debug tracing

Explanation
See message EZYRC04I.

System action
OREXEC ends.

Operator response
By examining the command typed or using the -d option the problem should be apparent.

System programmer response
None.
Module
rexec1.c

Procedure name
usage()

**EZYRC08I**
-l *usr* specifies remote login id

Explanation
See message EZYRC04I.

System action
OREXEC ends.

Operator response
By examining the command typed or using the -d option the problem should be apparent.

System programmer response
None.

Module
rexec1.c

Procedure name
usage()

**EZYRC09I**
-p *pwd* specifies remote password

Explanation
See message EZYRC04I.

System action
OREXEC ends.

Operator response
By examining the command typed or using the -d option the problem should be apparent.

System programmer response
None.

Module
rexec1.c

Procedure name
usage()

**EZYRC10I**
-s *port* specifies server port
Explanation
See message EZYRC04I.

System action
OREXEC ends.

Operator response
By examining the command typed or using the -d option the problem should be apparent.

System programmer response
None.

Module
rexec1.c

Procedure name
usage()

EZYRC11I -V display APAR level

Explanation
See message EZYRC04I.

System action
OREXEC ends.

Operator response
By examining the command typed or using the -d option the problem should be apparent.

System programmer response
None.

Module
rexec1.c

Procedure name
usage()

EZYRC12I Example: orexec -d -l guest -p guest hostname ls -l

Explanation
See message EZYRC04I.

System action
OREXEC ends.
Operator response
By examining the command typed or using the -d option the problem should be apparent.

System programmer response
None.

Module
rexec1.c

Procedure name
usage()

EZYRC13I XXX YYY

Explanation
XXX is the program called (MVS OREXEC).
YYY is the APAR number, or if there have been no APARs applied, it is base.

System action
OREXEC continues.

Operator response
None.

System programmer response
None.

Module
rexec1.c

Procedure name
main()

EZYRC14I -s port fhost command

Explanation
This the second of a group of messages that is issued when a question mark has been entered, or an invalid number of arguments have been entered.

System action
OREXEC ends.

Operator response
Correct the invalid argument and try the command again.

System programmer response
None.
Module
rexec1.c

Procedure name
usage()

EZYRC15I     -C Uppercase messages

Explanation
This is part of a group of messages that is issued when a question mark has been entered or an invalid number of arguments have been entered.

System action
OREXEC ends.

Operator response
Correct the invalid argument and try the command again.

System programmer response
None.

Module
rexec1.c

Procedure name
usage()

EZYRC16E     function failed. description rsn = errnojr

Explanation
An unexpected value was returned from the function indicated by the function value. This message can be issued from TSO REXEC or orexec. Processing ends.

function is the name of the C/C++ run-time function. See the z/OS XL C/C++ Runtime Library Reference for information about C/C++ run-time functions.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
OREXEC ends.

Operator response
Reissue the OREXEC command, if the problem persist contact the system programmer.

System programmer response
Correct the error indicated by description and errnojr.
Module
EZATREXE or EZARCC1C

Procedure name
main()

EZYRC17I Select time limit expired.

Explanation
The time limit on the select expired before a response was received from the server. Processing ends.

System action
OREXEC ends.

Operator response
Attempt the OREXEC command again.

System programmer response
None.

Module
rexc1.c

Procedure name
main()

EZYRC18I Foreign host name is missing.

Explanation
The foreign host name was not specified on the OREXEC command. It is required. Processing ends.

System action
OREXEC ends.

Operator response
Attempt the OREXEC command again with the foreign host name.

System programmer response
None.

Module
rexc1.c

Procedure name
main()

EZYRC19I Data socket = dsock, Control socket = csock.
**Explanation**
This is a debug message indicating the socket descriptors being used by the z/OS UNIX REXEC client. It is issued only when the -d option is specified on the OREXEC command.

**System action**
OREXEC continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
rexec1.c

**Procedure name**
main()

**EZYRC23E** Command is missing

**Explanation**
A command was not specified on the OREXEC command.

**System action**
OREXEC ends.

**Operator response**
Reissue the OREXEC command and specify a command.

**System programmer response**
None.

**Module**
rexec1.c

**Procedure name**
main()

**EZYRC24I** -e wait select time limit

**Explanation**
See message EZYRC04I.

**System action**
OREXEC ends.
Operator response
Examine the command typed or use the -d option to diagnose the problem.

System programmer response
None.

Module
reexec1.c

Procedure name
usage()

EZYRC31I Calling function rcmd_af with the following:

Explanation
This message is followed by EZYRC02I, which lists the host, user, cmd, and port that are passed to the local rcmd_af function, which passes the rsh information to the server.

System action
ORSH continues.

Operator response
None.

System programmer response
None.

Module
ussrsh.c

Procedure name
main()

EZYRC33E The call to rcmd_af procedure failed: description rsn = errnojr

Explanation
The call to the rcmd_af procedure that will pass the information to the remote host failed.
description describes the error.
errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
ORSH ends.
**Operator response**
Examine the command typed and reissue the ORSH command. If the problem persists, contact the system programmer. The most common description is ECONNREFUSED. If this occurs, start the remote shell server on the remote host.

**System programmer response**
Correct the error indicated by *description* and *errnojr*.

**Module**
ussrsh.c

**Procedure name**
main()

---

**EZYRD01**
Invalid option in `/etc/inet.conf`

**Explanation**
An option that is not valid has been used in the `/etc/inetd.conf` file. The allowable options are: dIv

**Note:** If an invalid option has been specified, none of the valid options will be in effect.

**System action**
REXEC continues.

**Operator response**
None.

**System programmer response**
Correct the configuration options in `/etc/inetd.conf` for the entry exec under the column service name.

**Module**
rexecd.c

**Procedure name**
main

---

**EZYRD02E**
getpeername failure

**Explanation**
Getpeername could not return the name of the peer that is connected to the socket.

**System action**
REXEC ends.

**Operator response**
Try the command again later.
**System programmer response**

See message EZYRD33E on the host where rexecd is running for more information including the failure description (errno) and reason code (errnojr).

**Module**

rexecd.c

**Procedure name**

main

**EZYRD03I  Remote address = XX.XX.XX.XX**

**Explanation**

XX.XX.XX.XX is the IPv4 or IPv6 address that is the internet address that the login ID connected from. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

**System action**

REXEC continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

rexecd.c

**Procedure name**

main

**EZYRD04E  Binary one not received.**

**Explanation**

The first byte of the string that is passed to the server is supposed to be a binary 1. This did not happen in this case.

**System action**

REXEC ends.

**Operator response**

Try the command again.

**System programmer response**

The string that is being passed from the client is not in the proper format. Check the string that is being passed from the client.
Module
rexecd.c

Procedure name
doit()

EZYRD05I  clisecport = n

Explanation
n is the number of the secondary port that is being passed in from the client. This is the port that rexecd will send the output to. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

System action
REXECD continues.

Operator response
None.

System programmer response
None.

Module
rexecd.c

Procedure name
doit()

EZYRD06E  Unable to create secondary port.

Explanation
The socket subroutine was unable to create a socket in the specified AddressFamily and of the specified type.

System action
REXECD ends.

Operator response
Try the command again.

System programmer response
The socket could not be created for one of the following reasons:
• The addresses in the specified address family cannot be used with this socket.
• The socket in the specified address family is not supported.
• The per-process descriptor table is full.
• Insufficient resources were available in the system to complete the call.
Module
rexecd.c

Procedure name
doit()

EZYRD07E   Cannot make second port

Explanation
The rexecd server was unable to connect to the specified port.

System action
REXECDC ends.

Operator response
Try the command again.

System programmer response
The connect socket call to connect with the requested socket on the client failed for one of the following reasons:

• The Socket parameter is not valid.
• The Socket parameter refers to a file, not a socket.
• The specified address is not available from the local machine.
• The addresses in the specified address family cannot be used with this socket.
• The socket is already connected.
• The establishment of a connection timed out before a connection was made.
• The attempt to connect was rejected.
• No route to the network or host is present.
• The specified address is already in use.
• The Address parameter is not in a writable part of the user address space.
• The socket is marked as nonblocking. The connection cannot be immediately completed. The application program can select the socket for writing during the connection process.
• The specified path name contains a character with the high-order bit set.

Module
rexecd.c

Procedure name
doit()

EZYRD08I   User is: XXX

Explanation
XXX is the login ID that has been passed from the remote host. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.
System action
REXECD continues.

Operator response
None.

System programmer response
None.

Module
rexecd.c

Procedure name
doit()

EZYRD09I Command is: XXX

Explanation
XXX is the command that has been passed from the remote host. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

System action
REXECD continues.

Operator response
None.

System programmer response
None.

Module
rexecd.c

Procedure name
doit()

EZYRD10E XXX: unknown login. cmd = YYY

Explanation
XXX is the logon ID that was passed in from the remote host.
YYY is the command that was passed in.

System action
REXECD ends.

Operator response
Correct the login ID and try again.
System programmer response
This message is written to the file that is specified by syslog.conf for information and authorization messages. It is only written out if either information or authorization is in the syslogd.conf file. This indicates that the login ID was not in the password structure.

Module
rexecd.c

Procedure name
doit()

EZYRD11E    Login incorrect.

Explanation
The login was incorrect.

System action
REXECD ends.

Operator response
Try the command again.

System programmer response
It is considered to be a security violation to tell the client if the user ID or password is incorrect so this is a generic message saying that one of them is incorrect.

Module
rexecd.c

Procedure name
doit()

EZYRD12I    Name is: XXX, user is YYY

Explanation
This is a debug message. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.
XXX is the login ID that has been passed in from the client.
YYY is the login ID as gotten from the password structure.

System action
REXECD continues.

Operator response
None.

System programmer response
None.
Module
rexecd.c

Procedure name
doit()

EZYRD13I dir is XXX

Explanation
XXX is the home directory that is gotten from the password structure. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

System action
REXEDC continues.

Operator response
None.

System programmer response
None.

Module
rexecd.c

Procedure name
doit()

EZYRD14I uid is: X, gid is Y

Explanation
X is the uid that is obtained from the password structure and Y is the gid that is obtained from the password structure. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

System action
REXEDC continues.

Operator response
None.

System programmer response
None.

Module
rexecd.c
**Procedure name**
doit()

**EZYRD15E**   XX wrong password, cmd = YY

**Explanation**
XX is the login ID passed in from the client and YY is the command that is being attempted. The password that has been passed in is not correct. This message is written to the file that is specified by syslog.conf for error messages. It is only written out if error is set up to write out in syslogd.

**System action**
REXECD ends.

**Operator response**
Correct the password and try again.

**System programmer response**
It is considered to be a security violation to tell the client if the user ID or password is incorrect so a generic message saying that one of them is incorrect will be sent to the client.

**Module**
rexecd.c

**Procedure name**
doit()

**EZYRD17E**   XX no home directory. cmd = YY

**Explanation**
XX indicates the login ID and YY indicates the command that is being attempted.

**System action**
REXECD ends.

**Operator response**
Contact the system programmer

**System programmer response**
No home directory has been set up in the password structure for the login ID that is identified in the message. Add a home directory in the password structure for this login ID.

**Module**
rexecd.c

**Procedure name**
doit()

**EZYRD18E**   No remote directory.
**Explanation**
No home directory has been set up on the remote host for the logon ID.

**System action**
REXECD ends.

**Operator response**
Contact the system programmer at the remote site and have a home directory set up in the password structure.

**System programmer response**
Set up a home directory in the password structure.

**Module**
rexecd.c

**Procedure name**
doit()

---

**EZYRD19E**  Cannot make pipe1.

**Explanation**
Trying to make a pipe to go between the parent and the child process. This pipe could not be made.

**System action**
REXECD ends.

**Operator response**
Try the failing command again and if it still fails contact the system programmer.

**System programmer response**
The pipe could not be made for one of the following reasons:
- The FileDescriptor parameter points to a location outside of the allocated address space of the process.
- Two file descriptors are already open (OPEN_MAX).
- The system file table is full, or the device containing pipes has no free i-nodes.

**Module**
rexecd.c

**Procedure name**
doit()

---

**EZYRD20E**  Cannot make pipe2.

**Explanation**
Trying to make a pipe to go between the parent and the child process for the purpose of doing ASCII to EBCDIC conversion. This pipe could not be made.
System action
REXECD ends.

Operator response
Try the failing command again and if it still fails contact the system programmer.

System programmer response
The pipe could not be made for one of the following reasons:
- The FileDescriptor parameter points to a location outside of the allocated address space of the process.
- Two file descriptors are already open (OPEN_MAX).
- The system file table is full, or the device containing pipes has no free i-nodes.

Module
rexecd.c

Procedure name
doit()

EZYRD21E Cannot make pipe3.

Explanation
Trying to make a pipe to go between the parent and the child process for the purpose of doing ASCII to EBCDIC conversion. This pipe could not be made.

System action
REXECD ends.

Operator response
Try the failing command again and if it still fails contact the system programmer.

System programmer response
The pipe could not be made for one of the following reasons:
- The FileDescriptor parameter points to a location outside of the allocated address space of the process.
- Two file descriptors are already open (OPEN_MAX).
- The system file table is full, or the device containing pipes has no free i-nodes.

Module
rexecd.c

Procedure name
doit()

EZYRD22E Cannot fork; try again.

Explanation
The parent process is trying to fork off a child process. It has not been able to do this.
**System action**
REXECD ends.

**Operator response**
Try the failing command again and if it still fails contact the system programmer.

**System programmer response**
The fork could not be done for one of the following reasons:

- The total number of processes executing system-wide or by a single user would be exceeded, or the system does not have the resources necessary to create another process.
- There is not enough space for this process.

**Module**
rexecd.c

**Procedure name**
doit()

---

**EZYRD23E** error on read socket 0

**Explanation**
An attempt has made to read data on the stdin socket. This attempt has not been successful.

**System action**
REXECD ends.

**Operator response**
Try the failing command again.

**System programmer response**
A read from socket 0 has failed. It has failed to read for one of the following reasons:

- The FileDescriptor parameter is not a valid file descriptor open for reading.
- The file was marked for nonblocking I/O, and no data was ready to be read.
- A read was interrupted by a signal before any data arrived, and the signal handler was installed with an indication that subroutines are not to be restarted.
- An I/O error occurred while reading from the file system.
- The process is a member of a background process attempting to read from its controlling terminal, and either the process is ignoring or blocking the SIGTTIN signal or the process group has no parent process.

**Module**
rexecd.c

**Procedure name**
doit()

---

**EZYRD24I** XX cmd = YY
Explanation
If the -L option has been specified and if the syslog.conf file has been set up to send information or authorization messages to a file then this message will appear in the file.

System action
REXECD continues.

Operator response
None.

System programmer response
None.

Module
rexecd.c

Procedure name
doit()

EZYRD25E  setgid failed

Explanation
A set group ID subroutine failed.

System action
REXECD ends.

Operator response
Contact the system programmer.

System programmer response
The set group ID can fail for one of two reasons:
• The process does not have appropriate privileges to set the GID.
• The value of the GID parameter is incorrect.

Module
rexecd.c

Procedure name
doit()

EZYRD26E  initgroups failed

Explanation
An initgroups subroutine failed.
System action
REXECD ends.

Operator response
Contact the system programmer.

System programmer response
The subroutine can fail for the following reasons:
• The number of supplementary groups of the specified user plus the basegid group exceeds the maximum number of groups allowed, or an invalid user is specified.
• An MVS environmental or internal error occurred.
• The System authorization facility (SAF) had an error.
• The caller is not authorized, only authorized users are allowed to alter the supplementary group IDs list.

Module
rexecd.c

Procedure name
doit()

EZYRD27E  setuid failed

Explanation
A setuid failed.

System action
REXECD ends.

Operator response
Contact the system programmer.

System programmer response
A setuid failed. The following are the reasons that a setuid can fail:
• The process is currently not able to change UIDs.
• The value of uid is incorrect.
• The process does not have appropriate privileges to set the UID to uid.

Module
rexecd.c

Procedure name
doit()

EZYRD28E  Error from execl()

Explanation
An error has occurred in trying to open the shell process to do the requested command.
System action
REXECD ends.

Operator response
Try the command again. If the problem persists contact the system programmer.

System programmer response
An exccl can fail for the following reasons:

- The new process's combined argument list and environment list has more bytes than the system defined limit ARG_MAX.
- The process did not have appropriate permissions to run the specified file for one of the following reasons:
  - The process did not have permission to search a directory named in your path.
  - The process did not have execute permission for the file to be run.
  - The file to be run was not a regular file and the system cannot run files of its type.
- The new process image file has the appropriate permission and has a recognized format, but the system does not support execution of a file with this format.
- A loop exists in symbolic links.
- One or more pathname components in path or file does not exist.
- The new process image file has the appropriate access permission but is not in the proper format.
- The new process requires more memory than is permitted by the operating system.
- A directory component of path or file is not really a directory.

Module
rexecd.c

Procedure name
doit()

EZYRD29E  String too long

Explanation
A string that is being converted from ASCII to EBCDIC is too long.

System action
REXECD ends.

Operator response
Try the command again and if the problem persists contact the system programmer.

System programmer response
A string that is outside the permissible limits is being passed to REXECD. Check the parameters being passed and if everything looks correct contact the support center.

Module
rexecd.c
Procedure name
getstr()

EZYRD30I   usage: rexecd -dLV

Explanation
This message follows EZYRD01. It displays the valid options that are permitted in /etc/inetd.conf for the rexecd server.

System action
REXECD continues.

Operator response
None.

System programmer response
Correct the options specified in /etc/inetd.conf and from a superuser ID do a kill -1 pid number to cause inetd to reread the inetd.conf file.

Module
rexecd.c

Procedure name
usage()

EZYRD31I   XXX YYY

Explanation
XXX is the program called (MVS REXEC).
YYY is the APAR number, or if there have been no APARs applied, it is base.

System action
REXECD continues.

Operator response
None.

System programmer response
None.

Module
rexecd.c

Procedure name
main()

EZYRD32E   Cannot make second port
Explanation
The rexecd server was unable to connect to the specified port.

System action
REXECD ends.

Operator response
Try the command again.

System programmer response
The connect socket call to connect with the requested socket on the client failed for one of the following reasons:

- The Socket parameter is not valid.
- The Socket parameter refers to a file, not a socket.
- The specified address is not available from the local machine.
- The addresses in the specified address family cannot be used with this socket.
- The socket is already connected.
- The establishment of a connection timed out before a connection was made.
- The attempt to connect was rejected.
- No route to the network or host is present.
- The specified address is already in use.
- The Address parameter is not in a writable part of the user address space.
- The socket is marked as nonblocking. The connection cannot be immediately completed. The application program can select the socket for writing during the connection process.
- The specified path name contains a character with the high-order bit set.

Module
rexecd.c

Procedure name
doit()

EZYRD33E  function failed. description rsn = ernojr

Explanation
An unexpected value was returned from function.

description describes the error.

ernojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
REXECD ends.

Operator response
Reissue the REXEC command, if the problem persist contact the system programmer.
**System programmer response**
Correct the error indicated by *description* and *errnojr*.

**Module**
rexecd.c

**Procedure name**
main()

**EZYRD34I Stderr socket is socket**

**Explanation**
Shows the socket number used for the standard error connection. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

**System action**
REXEC continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
rexecd.c

**Procedure name**
doit()

**EZYRD35I Int(socket): 20-bytes hex data 10 char ascii**

**Explanation**
Shows the data being sent from the client to the server after the command has been read. *socket* is the socket from which the data was read. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

**System action**
REXEC continues.

**Operator response**
None.

**System programmer response**
None.
Module
rexecd.c

Procedure name
doit()

**EZYRD36I  Ont(socket): 20-bytes hex data 10 char ascii**

**Explanation**
Shows the data being sent from the server to the client after the command has been read. `socket` is the socket to which the data was written. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

**System action**
REXECID continues.

**Operator response**
None.

**System programmer response**
None.

Module
rexecd.c

Procedure name
doit()

**EZYRD37I  Signal sigval received on socket socket**

**Explanation**
Shows the value of a signal received from the client. This signal will be passed to the child process executing the command. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

**System action**
REXECID continues.

**Operator response**
None.

**System programmer response**
None.
**Procedure name**
doit()

**EZYRD38I**  
**EOF received on socket socket**

**Explanation**
Indicates a EOF was received on the socket. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

**System action**
REXECD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
rexecd.c

**Procedure name**
doit()

**EZYRD39I**  
**Child stderr socket is socket**

**Explanation**
Identifies the socket assigned in the parent process to the child standard error. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rexecd and debug is set up to write out in syslogd.

**System action**
REXECD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
rexecd.c

**Procedure name**
doit()

**EZYRD42I**  
**getsockname failure**
**Explanation**
Getsockname could not return the local address structure that is connected to the socket.

**System action**
REXECD ends.

**Operator response**
Try the command again later.

**System programmer response**
See message EZYRD33E on the host where rexed is running for more information including the failure description (errno) and reason code (errnojr.)

**Module**
rexecd.c

**Procedure name**
main

---

**EZYRP35I**  orpcinfo -d prognum versnum

**Explanation**
The command line syntax is displayed to indicate that the user invoked ORPCINFO with incorrect arguments.

**System action**
ORPCINFO exits.

**Operator response**
Reenter ORPCINFO with the correct control parameter syntax.

**System programmer response**
None.

**Module**
ORPCINFO

**Procedure name**
Usage()

---

**EZYRP36E**  Sorry. You are not root

**Explanation**
You must have root authority to use the -d option.

**System action**
The registration is not deleted.
Operator response  
None.

System programmer response  
Inform the system administrator of the problem.

Module  
orpcinfo

Procedure name  
stderr

EZYRP37E  orpcinfo: Could not delete registration for prog program version version

Explanation  
orpcinfo failed to receive a positive response from the portmapper to its request to delete the specified program. See accompanying message for reason.

System action  
The registration is not deleted.

Operator response  
None.

System programmer response  
Use other RPC options as appropriate. If necessary, inform the system administrator of the problem.

Module  
orpcinfo

Procedure name  
stderr

EZYRP52E  oportmap CALLIT: cannot fork

Explanation  
The portmapper was not able to fork as required to process a broadcast request.

System action  
The portmapper ignores the broadcast request and continues.

Operator response  
None.

System programmer response  
Inform the system administrator of the problem.
Module
oportmap

Procedure name
stderr

**EZYRP66E** enablecache: cache already enabled
The rpc udp cache has already been enabled.

Explanation

System action
The server continues.

Operator response

System programmer response
Revise the program without the redundant call to svcudp_enablecache().

Module
svc_upd.o

Procedure name
stderr

**EZYRP67E** enablecache: could not allocate cache

Explanation
The server was not able to allocate storage for its internal tables.

System action
The server continues without the benefit of the cache request.

Operator response
None.

System programmer response
Revise the program to fit in the available storage.

Module
svc_upd.o

Procedure name
stderr

**EZYRP68E** enablecache: could not allocate cache data

Explanation
The server was not able to allocate storage for its internal tables.
System action
The server continues without the benefit of the cache request.

Operator response
None.

System programmer response
Revise the program to fit in the available storage.

Module
svc_upd.o

Procedure name
stderr

EZYRP69E    enablecache: could not allocate cache fifo

Explanation
The server was not able to allocate storage for its internal tables.

System action
The server continues without benefit of the cache request.

Operator response
None.

System programmer response
Revise the program to fit in the available storage.

Module
svc_upd.o

Procedure name
stderr

EZYRP70E    cache_set: victim not found

Explanation
The server was not able to find a cache reply that its internal tables indicated was there.

System action
The server continues without the benefit of the cache request.

Operator response
None.
System programmer response
Inform the system administrator of the problem.

Module
csvc_upd.o

Procedure name
stderr

**EZYRP71E** cache_set: victim alloc failed

Explanation
The server was not able to allocate storage for its internal tables.

System action
The server continues without the benefit of the cache request.

Operator response
None.

System programmer response
Revise the program to fit in the available storage.

Module
csvc_upd.o

Procedure name
stderr

**EZYRP72E** cache_set: could not allocate new rpc_buffer

Explanation
The server was not able to allocate storage for its cached reply.

System action
The server continues without the benefit of the cache request.

Operator response
None.

System programmer response
Revise the program to fit in the available storage.

Module
csvc_upd.o
Procedure name
stderr

**EZYRS01I**  **XXX YYY**

**Explanation**

XXX is the program called (MVS RSHD).
YYY is the APAR number, or if there have been no APARs applied, it is base.

**System action**

RSHD continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

rshd.c

**Procedure name**

main()

**EZYRS02**  **Invalid option in /etc/inetd.conf**

**Explanation**

An invalid option has been used in the /etc/inetd.conf file. The allowable options are: dlV NOTE: If an invalid option has been specified None. of the valid options will be in effect.

**System action**

RSHD continues.

**Operator response**

None.

**System programmer response**

Correct the configuration options in /etc/inetd.conf for the entry shell under the column service name.

**Module**

rshd.c

**Procedure name**

main

**EZYRS03E**  **Getpeername failure.**
**Explanation**
Getpeername could not return the name of the peer that is connected to the socket.

**System action**
RSHD ends.

**Operator response**
Try the command again later.

**System programmer response**
The name of the peer that is connected to the socket could not be found for one of the following reasons:
- The argument is not a valid descriptor.
- The argument is a file, not a socket.
- The socket is not connected.
- Insufficient resources were available in the system to perform the operation.
- The name parameter pointed to memory not in a valid part of process address space.

**Module**
rshd.c

**Procedure name**
main

---

**EZYRS04**  Setsockopt (SO_KEEPALIVE) failed
W

**Explanation**
There was a failure in setting the socket keepalive option in setsockopt.

**System action**
RSHD continues.

**Operator response**
Try the command again.

**System programmer response**
There was an error in setting the socket keepalive option when using setsockopt. This error should not occur. Contact the IBM Software Support Center.

**Module**
rshd.c

**Procedure name**
main

---

**EZYRS05**  Setsockopt (SO_LINGER) failed
W

738  z/OS Communications Server: IP Messages: Volume 3 (EZY)
**Explanation**
There was a failure in setting the socket linger option in setsockopt.

**System action**
RSHD continues.

**Operator response**
Try the command again.

**System programmer response**
There was an error in setting the socket linger option when using setsockopt. This error should not occur. Contact the IBM support center.

**Module**
rshd.c

**Procedure name**
main

**EZYRS06E**  Malformed from address.

**Explanation**
The IP address that was received cannot be put in the correct network byte order.

**System action**
RSHD ends.

**Operator response**
Correct the address being sent in from the client.

**System programmer response**
None.

**Module**
rshd.c

**Procedure name**
doit

**EZYRS07**  Connection received using IP options (ignored)

**Explanation**
An attempt was made to put special IP options in the data stream. All special options are ignored.

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

System programmer response
None.

Module
rshd.c

Procedure name
doit

EZYRS08E  Setsockopt IP_OPTIONS NULL.

Explanation
An unsuccessful attempt was made to zero out the IP options.

System action
RSHD ends.

Operator response
None.

System programmer response
This error should not occur. Contact the IBM System Support Center.

Module
rshd.c

Procedure name
doit

EZYRS09E  Connection from XX on illegal port YY

Explanation
An attempt was made to connect to RSHD using a non-reserved port. The port numbers that are considered to be reserved are 0-1023. IP options.

System action
RSHD ends.

Operator response
The client should attempt the connection on a reserved port.

System programmer response
None.
Module
rshd.c

Procedure name
doit

EZYRS10E   Connection on illegal port.

Explanation
An attempt was made to connect to RSHD using a non-reserved port. The port numbers that are considered to be reserved are 0-1023. IP options.

System action
RSHD ends.

Operator response
The client should attempt the connection on a reserved port.

System programmer response
None.

Module
rshd.c

Procedure name
doit

EZYRS11E   Binary one not received.

Explanation
The first byte of the string that is passed to the server is supposed to be a binary 1. This did not happen in this case.

System action
RSHD ends.

Operator response
Try the command again.

System programmer response
The string that is being passed from the client is not in the proper format. Check the string that is being passed from the client.
**EZYRS12I**  clisecport = n

**Explanation**

n is the number of the secondary port that is being passed in from the client. This is the port that rshd will send the output to. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

**System action**

RSHD continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

rshd.c

**Procedure name**

doit()

---

**EZYRS13E**  Cannot get stderr port

**Explanation**

The server has attempted to open a reserved port to be used for standard error.

**System action**

RSHD ends.

**Operator response**

Try again later.

**System programmer response**

Do a netstat conn to determine if all reserved ports are in use.

**Module**

rshd.c

**Procedure name**

doit()

---

**EZYRS16E**  Second port not reserved

**Explanation**

The server has attempted to open a port that is not reserved.
System action
RSHD ends.

Operator response
None.

System programmer response
The port number that the client has requested that the server use is not a reserved port number. The client must request a reserved port number.

Module
rshd.c

Procedure name
doit()

EZYR518E  Cannot make second port

Explanation
The rshd server was unable to connect to the specified port.

System action
RSHD ends.

Operator response
Try the command again.

System programmer response
The connect socket call to connect with the requested socket on the client failed for one of the following reasons:

• The Socket parameter is not valid.
• The Socket parameter refers to a file, not a socket.
• The specified address is not available from the local machine.
• The addresses in the specified address family cannot be used with this socket.
• The socket is already connected.
• The establishment of a connection timed out before a connection was made.
• The attempt to connect was rejected.
• No route to the network or host is present.
• The specified address is already in use.
• The Address parameter is not in a writable part of the user address space.
• The socket is marked as nonblocking. The connection cannot be immediately completed. The application program can select the socket for writing during the connection process.
• The specified path name contains a character with the high-order bit set.

Module
rshd.c
EZYRS19E   Cannot make second port

Explanation
The rshd server was unable to connect to the specified port.

System action
RSHD ends.

Operator response
Try the command again.

System programmer response
The connect socket call to connect with the requested socket on the client failed for one of the following reasons:
- The Socket parameter is not valid.
- The Socket parameter refers to a file, not a socket.
- The specified address is not available from the local machine.
- The addresses in the specified address family cannot be used with this socket.
- The socket is already connected.
- The establishment of a connection timed out before a connection was made.
- The attempt to connect was rejected.
- No route to the network or host is present.
- The specified address is already in use.
- The Address parameter is not in a writable part of the user address space.
- The socket is marked as nonblocking. The connection cannot be immediately completed. The application program can select the socket for writing during the connection process.
- The specified path name contains a character with the high-order bit set.

Module
rshd.c

EZYRS20I   Could not look up address for XXX

Explanation
A gethostbyname or getaddrinfo was done of XXX and the system was unable to find the specified name. The failure could be because of one of the following reasons:
- The host specified by the Name parameter was not found.
- The local server did not receive a response from an authoritative server. Try again later.
- An unrecoverable error.
- The requested Name is valid but does not have an Internet address at the name server.
System action
RSHD ends.

Operator response
None.

System programmer response
None.

Module
RSHD.c

Procedure name
doit()

EZYRS21I  Remote user is: XXX

Explanation
XXX is the login ID of the user on the remote host. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.

Operator response
None.

System programmer response
None.

Module
RSHD.c

Procedure name
doit()

EZYRS22I  Local user is: XXX

Explanation
XXX is the login ID that has been passed from the remote host. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.
Operator response
None.

System programmer response
None.

Module
RSHD.c

Procedure name
doit()

EZYRS23I   Command is: XXX

Explanation
XXX is the command that has been passed from the remote host. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.

Operator response
None.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS24E   WWW@XXX as YYY: Unknown login. cmd = ZZZ

Explanation
The remote user WWW at the remote site XXX logging on as user ID YYY does not have a valid logon ID. The command ZZZ was specified.

System action
RSHD ends.

Operator response
None.

System programmer response
The user ID YYY will need to have a valid logon ID specified in the security database.
Module
rshd.c

Procedure name
doit()

EZYRS25E Unknown login.

Explanation
The login was incorrect.

System action
RSHD ends.

Operator response
Try the command again.

System programmer response
It is considered to be a security violation to tell the client if the user ID or password is incorrect, so this is a generic message saying that one of them is incorrect.

Module
rshd.c

Procedure name
doit()

EZYRS26E WWW@XXX as YYY: no home directory. cmd = ZZZ

Explanation
The remote user WWW at the remote site XXX logging on as user ID YYY does not have a home directory specified. The command ZZZ was specified.

System action
RSHD ends.

Operator response
None.

System programmer response
The user ID YYY will need to have a home directory specified in the security database.

Module
rshd.c

Procedure name
doit()

EZYRS27E No remote directory.
**Explanation**
No home directory has been set up on the remote host for the logon ID.

**System action**
RSHD ends.

**Operator response**
Contact the system programmer at the remote site and have a home directory set up in the security database.

**System programmer response**
Set up a home directory in the security database.

**Module**
RSHD.c

**Procedure name**
doit()

**EZYRS28E**  
WWW@XXX as YYY: permission denied. cmd = ZZZ

**Explanation**
The remote user WWW at the remote site XXX logging on as user ID YYY does not have a valid logon ID or password. The command ZZZ was specified

**System action**
RSHD ends.

**Operator response**
None.

**System programmer response**
The user ID YYY will need to have a logon ID of password specified in the security database.

**Module**
rshd.c

**Procedure name**
doit()

**EZYRS30E**  
Logins currently disabled.

**Explanation**
The systems administrator has disabled logins.

**System action**
RSHD ends.
Operator response
None.

System programmer response
A file in the /etc directory with the name nologin has been created. This disables all logins except superusers.

Module
rshd.c

Procedure name
doit()

EZYRS31E Cannot make pipe.

Explanation
Trying to make a pipe to go between the parent and the child process. This pipe could not be made.

System action
RSHD ends.

Operator response
Try the failing command again and if it still fails contact the system programmer.

System programmer response
The pipe could not be made for one of the following reasons:
- The FileDescriptor parameter points to a location outside of the allocated address space of the process.
- Two file descriptors are already open (OPEN_MAX).
- The system file table is full, or the device containing pipes has no free i-nodes.

Module
rshd.c

Procedure name
doit()

EZYRS32E Cannot make pipe1.

Explanation
Trying to make a pipe to go between the parent and the child process. This pipe could not be made.

System action
RSHD ends.

Operator response
Try the failing command again and if it still fails contact the system programmer.
System programmer response
The pipe could not be made for one of the following reasons:

- The FileDescriptor parameter points to a location outside of the allocated address space of the process.
- Two file descriptors are already open (OPEN_MAX).
- The system file table is full, or the device containing pipes has no free i-nodes.

Module
rshd.c

Procedure name
doit()

EZYRS33E Cannot make pipe2.

Explanation
Trying to make a pipe to go between the parent and the child process. This pipe could not be made.

System action
RSHD ends.

Operator response
Try the failing command again and if it still fails contact the system programmer.

System programmer response
The pipe could not be made for one of the following reasons:

- The FileDescriptor parameter points to a location outside of the allocated address space of the process.
- Two file descriptors are already open (OPEN_MAX).
- The system file table is full, or the device containing pipes has no free i-nodes.

Module
rshd.c

Procedure name
doit()

EZYRS34E Cannot fork; try again.

Explanation
The parent process is trying to fork off a child process. It has not been able to do this.

System action
RSHD ends.

Operator response
Try the failing command again and if it still fails contact the system programmer.
System programmer response
The fork could not be done for one of the following reasons:

- The total number of processes executing system-wide or by a single user would be exceeded, or the system does not have the resources necessary to create another process.
- There is not enough space for this process.

Module
rshc.c

Procedure name
doit()

EZYRS35E

Explanation
An error occurred while reading data from socket zero. Socket zero is the standard input socket.

System action
RSHD ends.

Operator response
Try the failing command again and if it still fails, contact the system programmer.

System programmer response
This error should not occur. Contact the IBM System Support Center.

Module
rshc.c

Procedure name
doit()

EZYRS36I WWW@XXX as YYY: cmd = zzz

Explanation
If the -L option has been specified and if the syslog.conf file has been set up to send information or authorization messages to a file then this message will appear in the file.

System action
RSHD continues.

Operator response
None.

System programmer response
None.
Module
rshd.c

Procedure name
doit()

EZYRS37E  setgid failed

Explanation
A set group ID subroutine failed.

System action
RSHD ends.

Operator response
Contact the system programmer.

System programmer response
The set group ID can fail for one of two reasons:
• The process does not have appropriate privileges to set the GID.
• The value of the GID parameter is incorrect.

Module
rshd.c

Procedure name
doit()

EZYRS38E  initgroups failed

Explanation
An initgroups subroutine failed.

System action
RSHD ends.

Operator response
Contact the system programmer.

System programmer response
The subroutine can fail for the following reasons:
• The number of supplementary groups of the specified user plus the basegid group exceeds the maximum number of groups allowed, or an invalid user is specified.
• An MVS environmental or internal error occurred.
• The System authorization facility (SAF) had an error.
• The caller is not authorized, only authorized users are allowed to alter the supplementary group IDs list.
**EZYRS39E**  
**setuid failed**

**Explanation**  
A setuid failed.

**System action**  
RSHD ends.

**Operator response**  
Contact the system programmer.

**System programmer response**  
A setuid failed. The following are the reasons that a setuid can fail:

- The process is currently not able to change UIDs.
- The value of uid is incorrect.
- The process does not have appropriate privileges to set the UID to uid.

---

**EZYRS40E**

**Explanation**  
An error has occurred in trying to open the shell process to do the requested command.

**System action**  
RSHD ends.

**Operator response**  
Try the command again. If the problem persists contact the system programmer.

**System programmer response**  
An execl can fail for the following reasons:

- The new process's combined argument list and environment list has more bytes than the system defined limit ARG_MAX.
- The process did not have appropriate permissions to run the specified file for one of the following reasons:
  - The process did not have permission to search a directory named in your path.
  - The process did not have execute permission for the file to be run.
– The file to be run was not a regular file and the system cannot run files of its type.
• The new process image file has the appropriate permission and has a recognized format, but the system does not support execution of a file with this format.
• A loop exists in symbolic links.
• One or more pathname components in path or file does not exist.
• The new process image file has the appropriate access permission but is not in the proper format.
• The new process requires more memory than is permitted by the operating system.
• A directory component of path or file is not really a directory.

**Module**

rshd.c

**Procedure name**

doit()

<table>
<thead>
<tr>
<th>EZYRS41E</th>
<th>String too long</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation</strong></td>
<td>A string that is being converted from ASCII to EBCDIC is too long.</td>
</tr>
<tr>
<td><strong>System action</strong></td>
<td>RSHD ends.</td>
</tr>
<tr>
<td><strong>Operator response</strong></td>
<td>Try the command again and if the problem persists contact the system programmer.</td>
</tr>
<tr>
<td><strong>System programmer response</strong></td>
<td>A string that is outside the permissible limits is being passed to RSHD. Check the parameters being passed and if everything looks correct contact the support center.</td>
</tr>
</tbody>
</table>

**Module**

rshd.c

**Procedure name**

gestr()  

<table>
<thead>
<tr>
<th>EZYRS42E</th>
<th>rshd -adlnLV</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation</strong></td>
<td>This message follows EZYRS02W. It displays the valid options that are permitted in /etc/inetd.conf for the rshd server.</td>
</tr>
<tr>
<td><strong>System action</strong></td>
<td>RSHD ends.</td>
</tr>
<tr>
<td><strong>Operator response</strong></td>
<td>None.</td>
</tr>
</tbody>
</table>
**System programmer response**
Correct the options specified in /etc/inetd.conf and from a superuser ID do a kill -1 *pid number* to cause inetd to reread the inetd.conf file.

**Module**
rshd.c

**Procedure name**
getstr()

---

**EZYRS43E**  WWW as YYY: Unknown login. cmd = ZZZ

**Explanation**
The remote user WWW is logging on as user ID YYY does not have a valid logon ID. The command ZZZ was specified.

**System action**
RSHD ends.

**Operator response**
None.

**System programmer response**
The user ID YYY will need to have a valid logon ID specified in the security database.

---

**Module**
rshd.c

**Procedure name**
doit()

---

**EZYRS44E**  WWW as YYY: no home directory. cmd = ZZZ

**Explanation**
The remote user WWW logging on as user ID YYY does not have a home directory specified. The command ZZZ was specified.

**System action**
RSHD ends.

**Operator response**
None.

**System programmer response**
The user ID YYY will need to have a home directory specified in the security database.

---

**Module**
rshd.c
**Procedure name**
doit()

**EZYRS45E** WWW as YYY: permission denied. cmd = ZZZ

**Explanation**
The remote user WWW logging on as user ID YYY does not have a valid logon ID or password. The command ZZZ was specified.

**System action**
RSHD ends.

**Operator response**
None.

**System programmer response**
The user ID YYY will need to have a logon ID of password specified in the security database.

**Module**
rshd.c

**Procedure name**
doit()

**EZYRS46I** WWW as YYY: cmd = zzz

**Explanation**
If the -L option has been specified and if the syslog.conf file has been set up to send information or authorization messages to a file then this message will appear in the file. The remote user WWW logging on as user ID YYY has executed command zzzq.

**System action**
RSHD continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
rshd.c

**Procedure name**
doit()

**EZYRS47I** WWW as YYY: cmd = zzz
Explanation
If the -L option has been specified and if the syslog.conf file has been set up to send information or authorization messages to a file then this message will appear in the file. The remote user WWW logging on as user ID YYY has executed command zzz.

System action
RSHD continues.

Operator response
None.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS48I Trusted host activated

Explanation
The rsh client has issued a command that does not have a password in it. The trusted host installation exit has been invoked. This message is only written to the log file if syslogd.conf has been configured and LOG_INFO has been specified.

System action
RSHD continues.

Operator response
None.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS49I Trusted host authentication failed.

Explanation
The rsh client has issued a command that does not have a password in it. The trusted host installation exit has been invoked. The installation exit returned something other than a zero (0) return code. This message will be followed by message EZYRS36I. These messages are only written to the log file if syslogd.conf has been configured and LOG_INFO has been specified.
System action
RSHD ends.

Operator response
The .rhosts file has incorrect information or there has been an attempt to circumvent the authentication.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS50E  function failed. description rsn = errnojr

Explanation
An unexpected value was returned from function. Processing ends.

function is the name of the C/C++ run-time function. See the z/OS XL C/C++ Runtime Library Reference for information about C/C++ run-time functions.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
RSHD ends.

Operator response
Reissue the RSH command, if the problem persist contact the system programmer.

System programmer response
Correct the error indicated by description and errnojr.

Module
rshd.c

Procedure name
main()

EZYRS51I  Stderr socket is socket

Explanation
Shows the socket number used for the standard error connection. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.
System action
RSHD continues.

Operator response
None.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS52I  Int(socket): 20-bytes hex data 10 char ascii

Explanation
Shows the data being sent from the client to the server after the command has been read. socket is the socket from which the data was read. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.

Operator response
None.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS53I  Ont(socket): 20-bytes hex data 10 char ascii

Explanation
Shows the data being sent from the server to the client after the command has been read. socket is the socket to which the data was written. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.
Operator response
None.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS54I  Signal **signal** received on socket **socket**

Explanation
Shows the value of a signal received from the client. This signal will be passed to the child process executing the command. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.

Operator response
None.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS55I  EOF received on socket **socket**

Explanation
Indicates a EOF was received on the socket. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.

Operator response
None.
System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS56I  Child stdin socket is socket

Explanation
Identifies the socket assigned in the parent process to the child standard input. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.

Operator response
None.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

EZYRS57I  Child stdout socket is socket

Explanation
Identifies the socket assigned in the parent process to the child standard output. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.

Operator response
None.

System programmer response
None.
Module
rshd.c

Procedure name
doit()

**EZYRS58I**  Child stderr socket is socket

Explanation
Identifies the socket assigned in the parent process to the child standard error. This message is written to the file that is specified by syslog.conf for debug messages. It is only written out if both debug is turned on for rshd and debug is set up to write out in syslogd.

System action
RSHD continues.

Operator response
None.

System programmer response
None.

Module
rshd.c

Procedure name
doit()

**EZYRS59I**  Kerberos function funcname return code rcode

Explanation
This message is issued for debugging, error conditions, or both. It is written to syslog. It might be returned to the orsh client. If debugging is on for orshd, this message indicates that the Kerberos run-time function completed and gives the return code. If debugging is off for orshd, this message is issued if the Kerberos run-time function completes with a nonzero return code.

funcname is the Kerberos run-time function. Kerberos run-time functions are documented in the z/OS Integrated Security Services Network Authentication Service Administration.

rcode is the hexadecimal return code from the Kerberos run-time function. The return codes from the run-time functions are documented in the z/OS Integrated Security Services Network Authentication Service Administration under Kerberos Runtime Codes.

System action
orshd ends.

Operator response
If the message has a nonzero return code, the orshd client might not be authorized for kerberos use of orshd. Check with your system administrator.
System programmer response
Verify that the orsh client is authorized for Kerberos V5. Verify the Kerberos configuration.

Module
rshdkerb

Procedure name
do_kerb5()

EZYRS60E Kerberos error - errortype

Explanation
This message is issued to syslog when the orshd server detects an error in the Kerberos communication with the client.

errortype can be one of the following:

V1 validation
In the Kerberos exchange between client and server, the V1 validation failed.

V2 validation
In the Kerberos exchange between client and server, the V2 validation failed.

AP REQ validation
In the Kerberos exchange between client and server, the AP REQ validation failed.

checksum failed
The orshd server start up configuration options specify that the Kerberos checksum must be valid (using -m option). In the Kerberos exchange between client and server, the checksum was not valid.

cipher required
The orshd server start up configuration options specify that kerberos and encryption will be used (using -e option). In the exchange between client and server, the client did not specify encryption.

forwarding is unsupported
Credentials forwarding was requested by the orsh Kerberos client. The orshd server does not support credentials forwarding.

System action
orshd ends.

Operator response
Contact the system or network administrator.

System programmer response
Turn on debugging for orshd to determine where in the Kerberos flow the error occurred. Verify that the orsh client is a valid Kerberos V5 client. There is probably a mismatch in the Kerberos support or options supported between the orsh client and server. Review the Kerberos configuration between the client and the server.

Module
rshdkerb

Procedure name
do_kerb5()

EZYRS61E GSS-API function funcname error - errmsg
**Explanation**
This message is written to syslog by the orshd server. The Kerberos GSS-API run-time function failed. The text error message is provided.

`funcname` is the GSS-API run-time function that completed in error. GSS-API run-time functions are documented in the z/OS Integrated Security Services Network Authentication Service Programming.

`errmsg` is the text error message returned by the GSS-API run-time library. The messages from the run-time functions are documented in the z/OS Integrated Security Services Network Authentication Service Administration.

**System action**
orshd ends.

**Operator response**
The orshd client might not be authorized for GSS-API use of orshd. Contact your system administrator.

**System programmer response**
Turn on rshd debugging to determine where in the flow the error occurred. Verify that the client supports and is authorized for GSS. Verify the Kerberos GSS configuration between the client and server.

**Module**
rshdgss

**Procedure name**
do_gssapi()

**EZYRS62E**  GSS-API function CALLING error `call_error` ROUTINE error `routine_error`

**Explanation**
This orshd message accompanies EZYRS61E. When a GSS-API run-time function has a nonzero return code, the message is written to syslog. This message identifies the GSS-API CALLING error and the ROUTINE error as defined by the GSS-API, and might be used for debugging.

`call_error` is the calling error part of the major status value returned by the GSS-API run-time library function. The major status value is described in the z/OS Integrated Security Services Network Authentication Service Programming.

`routine_error` is the routine error part of the major status value returned by the GSS-API run-time library function. The major status value is described in the z/OS Integrated Security Services Network Authentication Service Programming.

**System action**
orshd ends.

**Operator response**
The orshd client might not be authorized for GSS-API use of orshd. Contact your system administrator.

**System programmer response**
Verify that the client supports and is authorized for GSS. Verify the Kerberos GSS configuration between the client and server.
**Module**

rshdgss

**Procedure name**
do_gssapi()

**EZYRS63I**  GSS-API function *funcname* complete

**Explanation**

This is an orshd informational message written to syslog if orshd has debugging turned on. It is used to determine which of the GSS-API run-time functions have completed successfully.

*funcname* is the GSS-API run-time function that completed. GSS-API run-time functions are documented in the z/OS Integrated Security Services Network Authentication Service Programming.

**System action**

None.

**Operator response**

None.

**System programmer response**

None.

---

**Module**

rshdgss

**Procedure name**
do_gssapi()

**EZYRS64I**  GSS-API received a token of length *token_length*

**Explanation**

This is an orshd GSS-API informational message written to syslog. It is a progress message for debugging and indicates that the orshd server received the GSS-API token from the client.

*token_length* is the decimal length of the GSS-API token received from the orsh client.

**System action**

None.

**Operator response**

None.

**System programmer response**

None.
Procedure name

do_gssapi()

EZYRS65I  GSS-API sent token of length token_length

Explanation
This is an orshd GSS-API informational message written to syslog. It is a progress message for debugging and indicates that the orshd server sent the GSS-API token to the client.

$token_length$ is the decimal length of the GSS-API token sent to the orsh client.

System action
None.

Operator response
None.

System programmer response
None.

Module
rshdgss

Procedure name
do_gssapi()

EZYRS66  -k start option not valid for IPv6 connection

Explanation
Kerberos version 5.0 or the GSSAPI is not supported for IPv6 connections.

System action
Connection to the client is closed.

Operator response
If tcp6 is specified in the inetd configuration file for the shell service, then start RSHD without the -k start option.

System programmer response
None.

Module
rshd.c

Procedure name
main()
Chapter 12. EZYTxxx messages

**EZYTE01E**  Cannot translate current code page.

**Explanation**
This message is written to the client as well as the syslog file. An invalid return code was received from the routine initxlate(), using the default code page ISO8859-1. The connection ends. This could be the result of one of the following conditions:

- Invalid return from iconv_open() for building either the ASCII EBCDIC translation table or the EBCDIC to ASCII translation table.
- Invalid return from iconv() after opening the translate table.

**System action**
Processing ends.

**Operator response**
Look for further information regarding the failure in the syslog file.

**System programmer response**
Correct the problem with the translation tables and reexecute the job.

**Module**
telnetd.c

**Procedure name**
main

**EZYTE02E**  Invalid suboption in /etc/inetd.conf. Option = option

**Explanation**
An invalid suboption has been specified in the /etc/inetd.conf file for -a, -c, -D, or -X option.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Correct the configuration options in /etc/inetd.conf for the entry otelnetd under the column service name.

**Module**
telnetd.c
Procedure name
main

**EZYTE03E**  Invalid option in /etc/inetd.conf. Option = *option*

**Explanation**
An invalid option has been specified in the /etc/inetd.conf file for telnetd.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Correct the configuration options in /etc/inetd.conf for the entry otelnetd under the column service name.

**Module**
telnetd.c

Procedure name
main

**EZYTE04I**  catgets description rsn = *errnojr*

**Explanation**
This describes an error which occurred with the first attempt to retrieve a message from the message catalog. If this message appears, otelnetd will use the default messages hard-coded within the software. Otherwise, otelnetd will be use the message catalog stored in the file /usr/lib/nls/msg/C/tnmsgs.cat.

*description* describes the error.

*errnojr* is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
telnetd.c

Procedure name
main
Explanation
If -D report/all is specified, the above messages will be issued. They describe the parameter settings after the /etc/inetd.conf for telnet is read and processed. Trace flag corresponds to -t. Debug flag corresponds to -D. linemode flags corresponds to -l. Kludgelinemode corresponds to -k. Keepalives correspond to -n. hostinfo flags correspond to -h and Registered host - U. multi_proc corresponds to -m.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
main

Explanation
This message is written to the client not the syslog file. It is informing the user that the parameters which were specified for telnet server are invalid.

System action
Processing continues.

Operator response
Check the /etc/inetd.conf for the parameters specified. Check the stderr file to determine which parameters were identified as invalid. The supported options will be printed in the syslog file. Correct the error and try again.

System programmer response
None.

Module
telnetd.c

Procedure name
usage

Usage: Supported options for telnetd:
Explanation
This message displays a list of all supported options for otelnetd. See z/OS Communications Server: IP Configuration Guide for more information.

System action
Processing ends.

Operator response
Log in again after the system programmer corrects the invalid options.

System programmer response
Correct the invalid options in /etc/inetd.conf.

Module
telnetd.c

Procedure name
usage

EZYTE09E Telnet session is ending.

Explanation
This message is written to the client as well as the syslog file. An error was found during processing the telnet server options.

System action
Processing ends.

Operator response
Correct the invalid option and re-execute the job.

System programmer response
None.

Module
telnetd.c

Procedure name
usage

EZYTE10I terminaltypeok: call tgetent (buf, terminal type)

Explanation
This message is issued if -t is specified for tracing. It identifies the terminal type which is being negotiated for this connection.

System action
Processing continues.
Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
terminalttypeok

EZYTE11I  doit: host_name host name doit: IP address IP_address doit: PORT port doit: host host_name

Explanation
The following are variables that are set as a result of the gethostname(), gethostbyaddr(), getnameinfo(), or getaddrinfo().

- **host_name** is the name server name related to the IP address of the client.
- **IP_address** is the Internet address of the client.
- **port** is the port that the client is using.
- **host_name** is the MVS system being telnetted to.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
doit

EZYTE12E  userid Permission denied.

Explanation
This message is written to the client as well as the syslog file. The security checks for the user ID specified failed.

System action
Processing ends.
Operator response
Verify that the user ID specified is valid for the system trying to access. Verify that the password is the correct password for this userid. Try the access again.

System programmer response
None.

Module
telnetd.c

Procedure name
doit

EZYTE13E  setgid failed description rsn = reason code

Explanation
This message is written to the client as well as the syslog file. A set group ID subroutine failed.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
Contact the system programmer.

System programmer response
The set group ID can fail for one of two reasons:
• The process does not have appropriate privileges to set the GID.
• The value of the GID parameter is incorrect.

Module
telnetd.c

Procedure name
doit()

EZYTE14E  initgroups failed description rsn = errnojr

Explanation
This message is written to the client as well as the syslog file. An initgroups subroutine failed.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.
System action
Processing ends.

Operator response
Contact the system programmer.

System programmer response
The subroutine can fail for the following reasons:
• The number of supplementary groups for the specified user plus the basegid group exceeds the maximum number of groups allowed, or an invalid user is specified.
• An MVS environmental or internal error occurred.
• The System authorization facility (SAF) had an error.
• The caller is not authorized, only authorized users are allowed to alter the supplementary group IDs list.

Module
telnetd.c

Procedure name
doit()

EZYTE15E  setuid failed description rsn = errnojr

Explanation
This message is written to the client as well as the syslog file. A setuid failed.
description describes the error.
errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Ernojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
Contact the system programmer.

System programmer response
A setuid failed. The following are the reasons that a setuid can fail:
• The process is currently not able to change UIDs.
• The value of uid is incorrect.
• The process does not have appropriate privileges to set the UID to uid.

Module
telnetd.c

Procedure name
doit()
uid is: uid, gid is gid

Explanation
X is the uid that is obtained from the password structure and Y is the gid that is obtained from the password structure. This message is written to the trace file if -t is specified as a telnet server option.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
doit()

Interrupt()

Explanation
Entered the interrupt subroutine. Used for tracing.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
interrupt()

sendbrk()

Explanation
Entered the sendbrk subroutine. Used for tracing.

System action
Processing continues.
Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
sendbrk()

EZYTE19I  sendsusp()

Explanation
Entered the sendsusp subroutine. Used for tracing.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
sendsusp()

EZYTE20I  recv_ayt()

Explanation
Entered the recv_ayt subroutine. Used for tracing.

System action
Processing continues.

Operator response
None.

System programmer response
None.
Module
telnetd.c

Procedure name
recv_ayt()

EZYTE21I  doeof()

Explaination
Entered the doeof subroutine. Used for tracing.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
doeof()

EZYTE22I  herald() entered for filename

Explanation
The herald routine was entered to process the file. This message is used for tracing.
In the message text:

filename
The file name that is being processed by the herald routine.

System action
Processing continues.

Operator response
None.

System programmer response
None.

User response
None.
**Problem determination**
Not applicable.

**Source**
z/OS Communications Server: z/OS UNIX Telnet server (otelnetd)

**Module**
telnetd.c

**Routing code**
*

**Descriptor code**
*

**Automation**
Not applicable for automation

**Example**
EZYTE22I herald() entered for /etc/otelnetd.banner

**Explanation**
This message is written to the client not the syslog file. A request from the server to the user to enter their userid.

**System action**
Processing continues.

**Operator response**
Specify the correct user ID for the system which you are telnet'ing to. Processing continues.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
verify_password()

**EZYTE28I userid Password:**

**Explanation**
This message is written to the client not the syslog file. A request from the server to the user to enter their password.
**System action**
Processing continues.

**Operator response**
Specify the correct password for the user ID previously issued. Processing continues.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
verify_password()

---

**EZYTE29I**  Starting new telnet session. catfd = catfd

**Explanation**
Informational message identifying the first message issued by the server when diagnostic processing is specified. It also outputs the file descriptor associated with the message catalog. If this value is -1, than the user will be using the default messages hard-coded within the software. Otherwise, the user will be accessing the message catalog stored in the /usr/lib/nls/msg/C/tnmsgs.cat.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
main

---

**EZYTE30E**  Invalid return code received from expired_pw().

**Explanation**
This message is written to the client as well as the syslog file. The password entered for the user ID specified has expired. The MVS SAF routine has been called to verify the new password entered. It complained resulting in the failure.

**System action**
Processing ends.
Operator response
Check for more information about the __passwd function in your syslog. Specify a valid new password and try again.

System programmer response
None.

Module
telnetd.c

Procedure name
verify_password()

EZYTE31E  Processing ends. Caller is not a member of BPX.DAEMON facility.

Explanation
This message is written to the client as well as the syslog file. The return from __passwd() function resulted in an errno of EPERM. This informs the user that they are not a member of the BPX.DAEMON security package.

System action
Error

Operator response
Contact your system support programmer to enable you for BPX.DAEMON

System programmer response
None.

Module
telnetd.c

Procedure name
verify_password()

EZYTE32W  You entered an invalid login name or password.

Explanation
This message is written to the client as well as the syslog file. The return from __passwd() function resulted in an errno stating that the password/user ID combination is invalid. The user will get three attempts to issue the correct password for the user ID specified. After the third invalid attempt the connection is ended.

System action
Processing continues until after the third attempt, then it is ended.

Operator response
Ensure you are issuing the correct password/user ID and try again.
System programmer response
None.

Module
telnetd.c

Procedure name
verify_password()

EZYTE33W  Password expired.

Explanation
This message is written to the client as well as the syslog file. The return from __passwd() function indicates that the password entered has expired.

System action
Processing continues.

Operator response
The program will continue processing and request that the user specify a new password.

System programmer response
None.

Module
telnetd.c

Procedure name
expired_pw()

EZYTE34I  Enter new password.

Explanation
This message is written to the client and not the syslog file. The return from __passwd() function indicates that the password entered has expired. A request for a new password is indicated.

System action
Processing continues.

Operator response
Enter new password.

System programmer response
None.

Module
telnetd.c
Procedure name
expired_pw()

EZYTE35I  Re-enter new password.

Explanation
This message is written to the client and not the syslog file. Reenter the password again to check for typos on entering the expired password.

System action
Processing continues.

Operator response
Enter new password.

System programmer response
None.

Module
telnetd.c

Procedure name
expired_pw()

EZYTE36E  You entered an invalid password.

Explanation
This message is written to the client as well as syslog file. A call was made to __passwd() to change the expired password. The new password specified resulted in errors.

System action
Processing continues.

Operator response
Check the errno and errnojr for more information. Enter new password.

System programmer response
None.

Module
telnetd.c

Procedure name
expired_pw()

EZYTE37E  New passwords do not match.
Explaination
This message is written to the client as well as syslog file. The second password entered does not match the first password entered.

System action
Processing continues.

Operator response
reenter new password.

System programmer response
None.

Module
telnetd.c

Procedure name
expired_pw()

EZYTE38E    Password too long.

Explaination
This message is written to the client as well as syslog file. The password specified is too long. Only a maximum of eight characters can be used for the password.

System action
Processing continues.

Operator response
Pick a new password.

System programmer response
None.

Module
telnetd.c

Procedure name
expired_pw()

EZYTE39W    Terminal type = DUMB. No full screen applications are supported

Explaination
This message is written to the syslog during diagnostic processing. Basically the terminal type specified during telnet negotiation was unsupported. Processing will continue but it will act as a dumb terminal. This means that all full screen applications will not work. Character-at-a-time raw mode applications such as vi will not work with a dumb terminal because they require full screen curson support. Terminal types identified as 3270 will be changed to DUMB terminals.
System action
Processing continues.

Operator response
Do not use any full screen application.

System programmer response
None.

Module
telnetd.c

Procedure name
getterminaltype

**EZYTE40E**
cleanup: waitpid failed. description rsn = errnojr

Explanation
The waitpid() provides a general interface that need to wait for certain child processes, that need resource utilization statistics accumulated by child process. A -1 was returned from the waitpid() for the child.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing continues.

Operator response
None.

System programmer response
The following conditions could be the result of an waitpid failure.

- The calling process has no existing unwaited-for child processes.
- The wait_status point to an illegal address.
- The call was interrupted by a caught signal or the signal did not have the SA_RESTART flag set.

Module
telnetd.c

Procedure name
cleanup()

**EZYTE41I**
cleanup: child exit status = wait status

Explanation
The result of the waitpid specified during cleanup.
**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
cleanup()

**EZYTE44E  Fork utmp error. description rsn = errno**

**Explanation**
This message is written to the client as well as the syslog file. The parent process is trying to fork off a child process. It has not been able to do this.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing ends.

**Operator response**
Try the failing command again and if it still fails contact the system programmer.

**System programmer response**
The fork could not be done for one of the following reasons:

- The total number of processes executing system-wide or by a single user would be exceeded, or the system does not have the resources necessary to create another process.
- There is not enough space for this process.

**Module**
telnetd.c

**Procedure name**
doit()

**EZYTE45E  parent: utmp error = wait status**

**Explanation**
The parent process is responding to the waitpid issued for the utmp processing. The parent could not cleanup the pid entry for the user ID specified.
**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Possibly the /etc/utmpx file had been updated already to remove the user ID associated with this pid number.

**Module**
telnetd.c

**Procedure name**
cleanup()

**EZYTE46E** 
**EXECL: error description**
**rn = errnojr**

**Explanation**
This message is written to the client as well as the syslog file. An error has occurred in trying to open the shell process to do the requested command. 

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing ends.

**Operator response**
Try the command again. If the problem persists contact the system programmer.

**System programmer response**
An execl can fail for the following reasons:

- The new process's combined argument list and environment list has more bytes than the system defined limit ARG_MAX.
- The process did not have appropriate permissions to run the specified file for one of the following reasons:
  - The process did not have permission to search a directory named in your path.
  - The process did not have execute permission for the file to be run.
  - The file to be run was not a regular file and the system cannot run files of its type.
- The new process image file has the appropriate permission and has a recognized format, but the system does not support execution of a file with this format.
- A loop exists in symbolic links.
- One or more pathname components in path or file does not exist.
- The new process image file has the appropriate access permission but is not in the proper format.
- The new process requires more memory than is permitted by the operating system.
• A directory component of path or file is not really a directory. Correct the error indicated by description and errnojr.

**Module**
telnetd.c

**Procedure name**
doit()

**EZYTE47I** Int: 20 bytes hex data 10 char ascii.

**Explanation**
This is hex/ascii representation of data which was inputted from the client to the telnet server via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
protocol.h

**Procedure name**
telnet()

**EZYTE48I** Ont: 20 bytes hex data 10 char ascii.

**Explanation**
This is hex/ascii representation of data which was outputted from the telnet server to the client via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
**Procedure name**
telnet()

**EZYTE49I** Ipt: 20 bytes hex data 10 char ascii.

**Explanation**
This is hex/ascii representation of data which was inputted from the child to the telnet server via the master file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D ptydata or -D all. There might be an additional 4 bytes which are control data which will not be displayed with this message.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
protocol.h

**Procedure name**
telnet()

**EZYTE51W** terminaltypeok: Tgetent failure

**Explanation**
The terminal type which was passed to the CURSES routine tgetent, did not match the supported types. Processing will attempt to negotiate a new terminal type. The supported terminal types can be found in /usr/lib/terminfo/ibm

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
terminaltypeok()

**EZYTE52E** Couldn't resolve your address into a host name. IP address is IP address description rsn = errnojr

EZYTxxxx messages 787
Explanation
This message is written to the client as well as the syslog file. The function call to a gethostbyaddr() or getnameinfo() resulted in a failure. gethostbyaddr() or getnameinfo() returns a pointer to an object describing an Internet host referenced by address. This structure contains the information obtained from the name server.

description describes the error.

ernojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends. if -U specified. Otherwise, processing continues.

Operator response
None.

System programmer response
Correct the error indicated by description and errnojr. The following are possible reasons for failure:

• No such host is known.
• The local server did not receive a response from an authoritative server. Try again at some later time.
• Some unexpected server failure was encountered.
• The requested name is valid but does not have an IP address associated with this name.
• The local name server might be down.

Module
telnetd.c

Procedure name
doit()

EZYTE54E All network ports in use.

Explanation
This message is written to the client as well as the syslog file. An attempt was made to identify ttys for the child process. An associated tty could not be obtained.

System action
Processing ends.

Operator response
Wait until some network ports are available.

System programmer response
Ensure that you have enough /dev/ptypXXXX and /dev/ttypXXXX for the number of users accessing your system. Examine the ports in use and ensure that none of them are zombies. Eliminate any zombies if found to free up some more ports.
Module
telnetd.c

Procedure name
doit()

EZYTE55E Getpty failed.

Explanation
An attempt was made to identify ttys for the child process. An associated tty could not be obtained.

System action
Processing ends.

Operator response
Wait until some network ports are available.

System programmer response
Ensure that you have enough /dev/ptypXXXX and /dev/ttypXXXX for the number of users accessing your system. Examine the ports in use and ensure that none of them are zombies. Eliminate any zombies if found to free up some more ports.

Module
telnetd.c

Procedure name
doit()

EZYTE56I Yes

Explanation
This message is written to the client not the syslog file. A request was issued for AYT by the user.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
recv_ayt()

EZYTE57I TELNETD: netwrite len chars
**Explanation**
The software issued data to be written to the client. This identifies the number of characters transmitted over the socket.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
recv_ayt

**EZYTE59I** read_pw: Character ignored char

**Explanation**
Telnet was reading in information typed in by the user (usually user ID or password). There is an extraneous byte of information received after the carriage return (new line). This byte is ignored and processing is continued.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
read_pw()

**EZYTE60I** cleanup(): child_pid = child_pid

**Explanation**
The child_pid specified is associated with the spawn issued at the start of this connection for the child.

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

System programmer response
None.

Module
telneth.c

Procedure name
cleanup()

EZYTE61E parent: waitpid failed, status = wait status

Explanation
The waitpid() provides a general interface that need to wait for certain child processes, that need resource utilization statistics accumulated by child process. A -1 was returned from the waitpid() for the child.

System action
Processing continues.

Operator response
None.

System programmer response
The following conditions could be the result of a waitpid failure.
• The calling process has no existing unwaited-for child processes.
• The wait_status point to an illegal address.
• The call was interrupted by a caught signal or the signal did not have the SA_RESTART flag set.

Module
telneth.c

Procedure name
cleanup()

EZYTE63I Int: 20 bytes hex data 10 char ascii.

Explanation
This is hex/ascii representation of data which was inputted from the client to the telnet server via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all. It corresponds to telnet negotiation data seen during processing of password and/or userid.

System action
Processing continues.

Operator response
None.
System programmer response
None.

Module
telnetd.c

Procedure name
read_pw()

EZYTE64I  Int: 20 bytes hex data 10 char ascii.

Explanation
This is hex/ascii representation of data which was inputted from the client to the telnet server via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all. It corresponds to input data received for password/user ID which exceeds the length of the buffer for password/user ID (8 bytes). This data will be ignored for further processing.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
read_pw()

EZYTE65I  PROTOCOL: send IAC Data Mark.

Explanation
A Data mark was sent to the client by the server.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
protocol.h
**Procedure name**
telnet()

**EZYTE66I**  
PROTOCOL: lmodetype = *lmodetype* linemode = *linemode* useline = *useline*

**Explanation**

*lmodetype* signifies whether the client handle real linemode or if use of kludgelinemode is needed. It will be set to one of the following:

- 0x04 REAL-LINEMODE - use the linemode option
- 0x03 KLUDGE-OK
- 0x02 NO-AUTOKLUDGE
- 0x01 KLUDGE-LINMEODE - use kludge linemode
- 0x00 NO-LINMODE - client is ignorant of linemode

*linemode* is true if linemode is currently on.

*useline* is the state that we wish to be in.

**System action**

Processing continues.

**Operator response**

None.

**System programmer response**

None.

**Module**

protocol.h

**Procedure name**
telnet()

**EZYTE67I**  
S(nfd): socketfd..ibits=ibits obits=obits ebits=ebits  
S(nfd): pty...ibits=ibits obits=obits ebits=ebits

**Explanation**

These messages are only issued if -t is specified in /etc/inetd.conf. They correlate to the conditions of the read, write and exceptions on either the masterfd (pty) or the socketfd. The are associated with what will happen within the protocol function. They determine if reading/writing of the master file descriptor or the socket file descriptor will occur. They will also determine if there is an exception outstanding on a particular pty which needs to be addressed.

**System action**

Processing continues.

**Operator response**

None.
System programmer response
None.

Module
protocol.h

Procedure name
telnet()

EZYTE68I  Ept: #bytes = cc pkcontrol(cntl) cntl

Explanation
These messages are only issued if -t is specified in /etc/inetd.conf. They correlate to the conditions of the exceptions on the pty. An exception condition was identified which needs to be processed. This identifies the number of bytes read in for the exception and what the actual exception condition was which needs to be handled.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
protocol.h

Procedure name
telnet()

EZYTE69I  PROTOCOL: cntl = cntl

Explanation
These messages are only issued if -t is specified in /etc/inetd.conf. They correlate to the conditions of the exceptions on the pty. The resulting exception condition after it has been processed with out-of-band data.

System action
Processing continues.

Operator response
None.

System programmer response
None.
Module
protocol.h

Procedure name
telnet()

**EZYTE70W**  \[\text{PROTOCOL}: \text{Found an unknown exception.}\]

**Explanation**
These messages are only issued if -t is specified in /etc/inetd.conf. They correlate to the conditions of the exceptions on the pty. An exception was read in which we are not processing currently. It is ignored. The ones currently handled are:

- FLUSHWRITE
- DOSTOP
- NOSTOP
- CHCP
- IOCTL

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

---

**Module**
protocol.h

**Procedure name**
telnet()

**EZYTE71I**  \[\text{SYNCHing is turned on}\]

**Explanation**
These messages are only issued if -t is specified in /etc/inetd.conf. They correlate to the conditions of the exceptions on the pty.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
Module
protocol.h

Procedure name
telnet()

EZYTE72I  PROTOCOL: netread ncc chars.

Explanation
These messages are only issued if -t is specified in /etc/inetd.conf. Identifies the number of bytes read in from the client to be either sent on to the child or processed by the server.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
protocol.h

Procedure name
telnet()

EZYTE73I  PROTOCOL: EOF on socket.

Explanation
These messages are only issued if -t is specified in /etc/inetd.conf. They correlate to the conditions of the exceptions on the pty.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
protocol.h

Procedure name
telnet()

EZYTE74I  PROTOCOL: EOF on master tty.
**Explanation**
These messages are only issued if -t is specified in /etc/inetd.conf. They correlate to the conditions of the exceptions on the pty. An EOF was received on the master tty. This results in the cleanup routine being called and the session eventually terminated.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
protocol.h

**Procedure name**
telnet()

---

**EZYTE78I** PROTOCOL: ptyread pcc chars

**Explanation**
These messages are only issued if -D report or -D netdata specified in /etc/inetd.conf. They correlate to the number of bytes read in from the child. They will be sent on to the client after processing by the server.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
protocol.h

**Procedure name**
telnet()

---

**EZYTE79E** PROTOCOL: select mask too small, increase FD_SETSIZE rsn = errno;
description

**Explanation**
This message is written to the client as well as the syslog file. Select uses bit masks of file descriptors in longs. FD_SETSIZE is defaulted to 2048.

description describes the error.
errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojr) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Increase the FD_SETSIZE to an appropriate number and rebuild the server.

**Module**
protocol.h

**Procedure name**
telnet()

---

**EZYTE80W**  
**PROTOCOL: select**  
description  
rsn = errnojr

**Explanation**
This message is written to the syslog file when the select() function call has terminated due to a timeout value being exceeded. The telnet session is ended.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojr) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Session is cancelled due to timeout.

**Operator response**
None.

**System programmer response**
Check the -c value in the /etc/inetd.conf file to verify that the timeout value is appropriate.

- The select() timed out.
- The telnet session is ended due to the select() timeout.
- Check the -c select timeout value in the /etc/inetd.conf file.

**Module**
protocol.h

**Procedure name**
telnet()

---

**EZYTE81E**  
**PROTOCOL: ioctl net FIONBIO.**  
description  
rsn = errnojr
Explanation
An unexpected value was returned from the ioctl.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing continues.

Operator response
None.

System programmer response
Correct the error indicated by description and errnojr. The following are some possible explanations of the failure:

- The file descriptor specified is invalid.
- The file descriptor specified is not associated with a character special device.
- The specified request does not apply to the kind of object that the socket file descriptor references.

Module
protocol.h

Procedure name
telnet()

EZYTE82E  PROTOCOL: error fcntl masterfd FIONBIO. description rsn = errnojr

Explanation
An unexpected value was returned from the ioctl.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing continues.

Operator response
None.

System programmer response
Correct the error indicated by description and errnojr. The following are some possible explanations of the failure:

- The file descriptor specified is invalid.
- The file descriptor specified is not associated with a character special device.
• The specified request does not apply to the kind of object that the socket file descriptor references.

Module
protocol.h

Procedure name
telnet()

EZYTE84I   PROTOCOL: SIMULATING receive

Explanation
An unexpected value was returned from the ioctl.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
protocol.h

Procedure name
telnet()

EZYTE85I   SYNCHing = SYNCHing

Explanation
An unexpected value was returned from the ioctl.

System action
Processing continues.

Operator response
None.

System programmer response
None.
EZYTE86W  TELNETD: setitimer description rsn = errnojr

Explanations
This message is written to the syslog file when the setitimer expires due to the timeout value being exceeded before the user had logged in. The telnet session is ended.
description describes the error.
errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Session is cancelled due to timeout.

Operator response
None.

System programmer response
Check the -c value in the /etc/inetd.conf file to verify that the timeout value is appropriate.
• The user timed out before logging in.
• The telnet session is ended due to the user timeout.
• Check the -c timeout value in the /etc/inetd.conf file.

Module
telnetd.c

Procedure name
telnet()

EZYTE87W  TELNETD: sigaction() failed - description rsn = errnojr

Explanations
This message is written to the syslog file when the sigaction() has failed. This will cause the user timeout value before login (-c option) not to be set. The telnet session continues without the timeout set.
description describes the error.
errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Signal handler for SIGALRM not set up, user timeout before login will not be set.

Operator response
None.

System programmer response
Correct the error indicated by description and errnojr.
**Module**
telnetd.c

**Procedure name**
telnet()

**EZYTE88E herald: function error on filename description rsn = errnojr**

**Explanation**
The herald routine attempted to perform the specified function using the specified file, but the function failed. The banner page from this file will not be displayed.

In the message text:

*function*  
The function call that failed.

*filename*  
The file name that is being processed by the herald routine.

*description*  
a description of the error.

*errnojr*  
The hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
Correct the error indicated by the *description* and *errnojr* values. If you do not want telnet to display this banner page, specify the -h option on the telnet configuration parameter.

**User response**
Not applicable.

**Problem determination**
See the system programmer response.

**Source**
z/OS Communications Server: z/OS UNIX Telnet server (otelnetd)
**Descriptor code**

*  

**Automation**

Not applicable for automation

**Example**

<table>
<thead>
<tr>
<th>EZYTE88E herald: stat error on /etc/otelnetd.banner  EDC5129I No such file or directory. rsn = 053B006C</th>
</tr>
</thead>
</table>

**EZYTE89I** herald: *filename* is an empty file

**Explanation**

The specified file is empty. The expected banner page is not displayed.

In the message text:

*filename*

  The name of the file that is being processed by the herald routine.

**System action**

Processing continues.

**Operator response**

None.

**System programmer response**

Ensure that a banner page is stored in the specified file, or if you do not want telnet to display this banner page, specify the `-h` option on the telnet configuration parameter.

**User response**

Not applicable.

**Problem determination**

Not applicable.

**Source**

z/OS Communications Server: z/OS UNIX Telnet server (otelnetd)

**Module**

telnetd.c

**Routing code**

*  

**Descriptor code**

*
Example

EZYTE89I herald: /etc/otelnetd.banner is an empty file

EZYTE90W Parameter -g ignored. Parameter not valid when -U coded

Explanation

This message is issued by z/OS UNIX Telnet server (otelnetd) if the -g parameter is specified when the -U parameter is also specified. The -g parameter causes otelnetd to disable gethostbyaddr and getnameinfo routines for the client IP address. The -U parameter causes otelnetd to drop connections from any IP address that cannot be mapped back into a symbolic name by the gethostbyaddr or getnameinfo routine. The -g parameter is ignored when the -U parameter is specified.

System action

Processing continues.

Operator response

Contact the system programmer.

System programmer response

Correct the conflicting parameters in the /etc/inetd.conf file.

User response

Not applicable.

Problem determination

Not applicable.

Source

z/OS Communications Server: z/OS UNIX Telnet server (otelnetd)

Module
telnetd.c

Routing code

*

Descriptor code

*

Automation

Not applicable for automation.

Example

EZYTE90W Parameter -g ignored. Parameter not valid when -U coded
EZYTO01I  Int: 20 bytes hex data 10 char ascii.

Explanation
This is hex/ascii representation of data which was inputted from the client to the telnet server via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all. It corresponds to input data received for password/user ID processing.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
read_pw()

EZYTO02I  Ont: 20 bytes hex data 10 char ascii.

Explanation
This is hex/ascii representation of data which was outputted from the client to the telnet server via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all. It corresponds to data received for password/user ID processing.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
read_pw()

EZYTO03E  Incoming session is not from a registered host.

Explanation
The server has been configured with a -U option which stipulates that only registered hosts will be accepted. The telnet session currently being processed is not registered and will be rejected.
System action
Processing ends.

Operator response
None.

System programmer response
Ensure that the incoming address is specified in the appropriate name server or etc/host file.

Module
telnetd.c

Procedure name
doit()

EZYTO04I   username = userid

Explanation
During processing of the userid/password, this message is issued if diagnostic processing is turned on. It will print out the user ID which was entered.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
verify_password()

EZYTO05I   Initial EBCDIC codepage = codepage, ascii codepage = codepage

Explanation
If diagnostic processing is turned on, this message will indicate what the initial code page settings are.

System action
Processing continues.

Operator response
None.
System programmer response
None.

Module
telnetd.c

Procedure name
main()

EZYTO06I Int: 20 bytes hex data 10 char ascii.

Explanation
This is hex/ascii representation of data which was outputted from the client to the telnet server via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all. It corresponds to data received for banner processing.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
herald()

EZYTO07I Ont: 20 bytes hex data 10 char ascii.

Explanation
This is hex/ascii representation of data which was outputted to the client from the telnet server via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all. It corresponds to data received for banner processing.

System action
Processing continues.

Operator response
None.

System programmer response
None.
Module
telnetd.c

Procedure name
herald()

EZYTO08E   Password too long.

Explanation
This message is written to the client as well as syslog file. The password specified is too long. Only a maximum of eight characters can be used for the password.

System action
Processing continues.

Operator response
Specify a correct password.

System programmer response
None.

Module
telnetd.c

Procedure name
read_pw()

EZYTO09I   options(entry) = value .

Explanation
This message is written to the syslog file. It identifies which options have been negotiated prior to the exec() taking place in the telnet server. The option array entry corresponds to a telnet option. The resulting value represents what was negotiated. If the value is 3 it correlates with the client has agreed to perform that function. If it is 12, then the server has agreed to perform this function.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c
EZYTO10E  Terminal type is not recognized. save_ttype

Explanation
The client responded to a DO TERMINAL TYPE, with a WILL TERMINAL TYPE. It then proceeded to send via subnegotiation, terminal types which are not supported by this server. As a valid terminal type was not received the session is ended.

System action
Processing ends

Operator response
Ensure that the client you are using can support terminal types which are supported by the curses function of tgetent(). At this time, this does not include 3270. Therefore, if your client is attempting to emulate a 3270 data stream the connection will not be accepted by this server.

System programmer response
None.

Module
telnetd.c

Procedure name
doit()

EZYTO11I  Int: 20 bytes hex data 10 char ascii.

Explanation
This is hex/ascii representation of data which was outputted from the client to the telnet server via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all. It corresponds to data received for telnet subnegotiation which was deferred until the tty was active.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
telnetd.c

Procedure name
undo_chars()
**EZYTO12E**  Logon user name too long.

**Explanation**
This message is written to the client as well as syslog file. The logon user name specified is too long. Only a maximum of eight characters can be used for the userid.

**System action**
Processing continues.

**Operator response**
Specify a correct userid.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
read_pw()

---

**EZYTS02I**  STATE:telrcv: send IAC c

**Explanation**
This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation SENT during the execution of the job.

**System action**
Telnet Server continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
state.c

**Procedure name**
telrcv

---

**EZYTS03E**  STATE:telrcv: panic state = state

**Explanation**
An IAC command is found during telnet negotiation and is followed by an unrecognized command option.
System action
Processing ends.

Operator response
None.

System programmer response
Issue traces to determine what was attempting to be negotiated as telnet control data. Trace to issue from server is '-D options'.

Module
state.c

Procedure name
telrcv

EZYTS04I  STATE:send_do: send option

Explanation
This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job. Specifically a request to issue a DO option.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
state.c

Procedure name
send_do

EZYTS05I  STATE:willoption: receive WILL option

Explanation
This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job. Specifically the receipt of WILL request.

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

**System programmer response**
None.

**Module**
state.c

**Procedure name**
willoption

**EZYTS06I**  STATE: willoption: set to kludge ok

**Explanation**
This is an informational message which is only issued if the user has specified -t in the /etc/inetd.conf file. It informs the user that kludgelinemode is operational.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
state.c

**Procedure name**
willoption

**EZYTS07I**  STATE:send_dont: send DON'T option

**Explanation**
This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job. Specifically a request to issue a DONT option.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.
Module:
state.c

Procedure name:
send_dont

EZYTS08I  STATE:wontoption: receive WON'T option

Explanation:
This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job. Specifically a response that a WONT was received.

System action:
Processing continues.

Operator response:
None.

System programmer response:
None.

Module:
state.c

Procedure name:
wontoption

EZYTS09I  STATE:send_will: send WILL option

Explanation:
This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job. Specifically a response that a WILL was sent.

System action:
Processing continues.

Operator response:
None.

System programmer response:
None.

Module:
state.c
**Explanation**

This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job. Specifically a response that a DO was received.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
state.c

**Procedure name**
dooption

---

**Explanation**

This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job. Specifically a response that a WONT was sent.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
state.c

**Procedure name**
send_wont
Explanation
This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job. Specifically a response that a DONT was received.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
state.c

Procedure name
dontoption

EZYTS13I  ENVIRON VALUE and VAR are reversed!

Explanation
This is an informational message which is only issued if the user has specified -D options in the /etc/inetd.conf file. This is issued during suboption negotiation. To be interoperable we need to determine if the VALUE and VAR values are reversed. If the first recognized character is a VAR or VALUE, then that will tell what type of client it is. If the first recognized character is a USERVAR, then we continue scanning the suboption looking for two consecutive VAR or VALUE fields. We should not get two consecutive VALUE fields. If a client has sent a well-formed option then the number of VALUES received should always be less than or equal to the number of VARS and USERVARs received. If not then the client has reversed the definitions.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
state.c

Procedure name
suboption

EZYTS14I  Opt: 20 bytes hex data 10 char ascii.
**Explanation**
This is hex/ascii representation of data which was Outputted from the client to the child via the master file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D ptydata or -D all.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
state.c

**Procedure name**
telrcv()

**EZYTS15I** STATE:dooption:deferred receive DO option

**Explanation**
This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job. Specifically a response that a DO was received prior to the creation of the tty which might require special tty processing.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
state.c

**Procedure name**
dooption

**EZYTS16I** STATE:willoption:deferred receive WILL option

**Explanation**
This is an informational message which is only issued if the user has specified -D options/report/all in the /etc/inetd.conf file. It informs the user of internal telnet negotiation occurring during the execution of the job.
Specifically a response that a WILL was received prior to the creation of the tty which might require special tty processing.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
state.c

**Procedure name**
willoption

---

**EZYTS17I**  STATE:Defer subotopn negotiation.

**Explanation**
This is an informational message that is only issued if the user has specified `-D options/report/all` in the `/etc/inetd.conf` file. Subnegotiation data was received that must be deferred until the tty is active. Now we can process this information.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
state.c

**Procedure name**
suboption()

---

**EZYTS18I**  STATE:Process deferred subotopn negotiation.

**Explanation**
This is an informational message which is only issued if the user has specified `-D options/report/all` in the `/etc/inetd.conf` file. Subnegotiation data was received that must be deferred until the tty is active. Now we can process this information.

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

System programmer response
None.

Module
state.c

Procedure name
suboption()

EZYTU01E  UTILITY:Read from ttloop. description rsn = errnojr

Explanation
This message is written to the client as well as the syslog file. The number of bytes returned from a read on the socketfd (client) is invalid. A negative value was received. The connection is ended.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
Correct the error indicated by description and errnojr.

System programmer response
None.

Module
utility.c

Procedure name
ttloop

EZYTU02E  UTILITY:Read from ttloop. Peer died. description rsn = errnojr

Explanation
This message is written to the client as well as the syslog file. The number of bytes returned from a read on the socketfd (client) is invalid. A value of zero was received. The connection is ended.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.
System action
Processing ends.

Operator response
Correct the error indicated by description and errnojr.

System programmer response
None.

Module
utility.c

Procedure name
ttloop

EZYTU03I UTILITY:ttloop read ncc chars.

Explanation
This message is issued if -D report or -D ptydata, is specified. It will write the number of characters that were read in from the socketfd, (client's socket) to be processed. It will be followed by hex data, preceded by the tag Int. This signifies a hex and ascii representation of input data coming into the telnet server from the network.

System action
Processing continues.

Operator response
None. Debug trace data.

System programmer response
None.

Module
utility.c

Procedure name
ttloop

EZYTU04E Telnetd: UTILITY: Can not translate current code page.

Explanation
This message is written to the client as well as the syslog file. An error was found while processing the initxlate() routine. The function will terminate the connection.

System action
Processing ends.

Operator response
None.
**System programmer response**
Check for additional messages issued.

**Module**
utility.c

**Procedure name**
ReturnToDefault

**EZYTU05E**  UTILITY: __tcsetcp. *description rsn = errnojr*

**Explanation**
This message is written to the client as well as the syslog file. An error was found while processing the default code pages.

*description* describes the error.

*errnojr* is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
Correct the error indicated by *description* and *errnojr*.

**Module**
utility.c

**Procedure name**
ReturnToDefault

**EZYTU06E**  Error using code pages newNames.__tccp_toname and newNames.__tccp_fromname - returning to default code pages.

**Explanation**
This message will be issued to the client and only to syslog file, if debug_mode is specified. The current code pages are not working correctly. Telnet is returning to the default code pages. If -D netdata is specified, this message will be followed by the hex and ascii translation of this message preceded by the tag 'Ont'.

**System action**
Processing ends.

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

Check the installation of the code pages. If requested code page is not valid, generate the appropriate code page. Otherwise, if the option is valid, check to see if code page is still good.

**Module**
utility.c

**Procedure name**
ReturnToDefault

**EZYTU07E**  
UTILITY: __tcgetcp. description rsn = errnojr

**Explanation**

This message is written to the client as well as the syslog file. An error was found while changing the code pages. 
*description* describes the error.

(errnojr) is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**

The connection is ended.

**Operator response**

None.

**System programmer response**

Correct the error indicated by *description* and *errnojr*.

**Module**
utility.c

**Procedure name**
change_translate

**EZYTU08I**  
Telnetd: UTILITY: Change to binary mode.

**Explanation**

The tty has been switched to binary mode.

**System action**

Processing continues.

**Operator response**

None.

**System programmer response**

None.
Module
utility.c

Procedure name
change_translate

EZYTU09I  UTILITY: Change to SingleByte pages newNames.__tccp_toname newNames.__tccp_fromname

Explanation
This is a single byte code page. A call to initxlate() will be made to setup the new code pages.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
change_translate

EZYTU10I  UTILITY: Change to MultiByte pages newNames.__tccp_toname newNames.__tccp_fromname

Explanation
This is a multi-byte code page. A call to doMultiByte() will be made to setup the new code pages.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
doMultiByte

EZYTU11E  UTILITY: Unexpected input string description rsn = errnojr
Explanation
This message is written to the client as well as the syslog file. The errno identified from iconv (on the ascii to EBCDIC converter) was EINVAL. EINVAL should only be encountered when the last character in the input buffer is incomplete. This did not occur.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
A2Emultiybyte_translate

EZYTU12E UTILITY: Unexpected iconv error description rsn = errnojr

Explanation
This message is written to the client as well as the syslog file. The errno identified from iconv (on the ascii to EBCDIC converter) is not currently checked for. Unexpected error detected.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
None.

System programmer response
None.

Module
utility.c
**Procedure name**
A2Emultiybyte_translate

**EZYTU13E**  UTILITY: stilloob: select *description rsn = errnojr*

**Explanation**
This message is written to the client as well as the syslog file. A check on the file descriptor was made to determine if out of band data existed. An invalid value was returned. The value was less than zero. 

*description* describes the error.

*errnojr* is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
None.

**Module**
utility.c

**Procedure name**
stilloob.

**EZYTU14I**  UTILITY: netwrite n chars.

**Explanation**
This message is issued if -D report or -D ptydata, is specified. It will write the number of characters which were read in from the socketfd, (client's socket) to be processed. It will be followed by hex data, preceded by the tag "Ont". This signifies a hex and ascii representation of Output data coming from the Telnet server to the net.

**System action**
Processing continues.

**Operator response**
None. Debug trace data.

**System programmer response**
None.

**Module**
utility.c
Procedure name
netflush

**EZYTU15I**  UTILITY: ptywrite n chars.

**Explanation**
This message is issued if -D report or -D ptydata, is specified. It will write the number of characters which were read in from the socketfd, (client's socket) to be processed. It will be followed by hex data, preceded by the tag "Opt". This signifies a hex and ascii representation of Output data coming from the Telnet server to the child.

**System action**
Processing continues.

**Operator response**
None. Debug trace data.

**System programmer response**
None.

**Module**
utility.c

Procedure name
ptyflush

**EZYTU16I**  UTILITY: Write of masterfd. description rsn = errnojr

**Explanation**
This message is issued if -D report or -t, is specified. A negative value was received of the number of bytes written to the masterfd. Processing continues and control is returned to the calling routine.

description describes the error.

eerrnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing continues.

**Operator response**
None. Debug trace data.

**System programmer response**
None.

**Module**
utility.c
Procedure name
ptyflush

EZYTU17I UTILITY: direction suboption sub_option

Explanation
A telnet negotiated suboption was either received or sent by the client.
direction is either send or receive.
sub_option is the Telnet negotiation suboption.

System action
Processing continues.

Operator response
None. Debug trace data.

System programmer response
None.

Module
utility.c

Procedure name
printsub

EZYTU18I UTILITY: write/send from NETFLUSH

Explanation
A telnet negotiated suboption was either received or sent by the client. The remaining portion of this command identifies what the suboption was.

System action
Processing continues.

Operator response
None. Debug trace data.

System programmer response
None.

Module
utility.c

Procedure name
printsub

EZYTU19E UTILITY: Unexpected iconv open error description rsn = errnoj

826  z/OS Communications Server: IP Messages: Volume 3 (EZY)
Explanation
The iconv_open failed for ascii to EBCDIC converter. The errno identified from iconv (on the ascii to EBCDIC converter) is not currently checked for. Unexpected error detected.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
None.

System programmer response
Ensure that the converter table being specified is valid and exists.

Module
utility.c

Procedure name
doMultiByte()

EZYTU20I  Int: 20 bytes hex data 10 char ascii.

Explanation
This is hex/ascii representation of data which was inputted from the client to the telnet server via the socket file descriptor. This is only seen during tracing of the server. The /etc/inetd.conf must have specified -D netdata or -D all.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
ttloop()

EZYTU21I  Ont: 20 bytes hex data 10 char ascii.
**Explanation**
This is hex/ascii representation of data which was outputted from the telnet server to the client via the socket file descriptor. This is only seen during tracing of the server. The `/etc/inetd.conf` must have specified `-D netdata` or `-D all.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
utility.c

**Procedure name**
netflush()

EZYTU22I   Opt: 20 bytes hex data 10 char ascii.

**Explanation**
This is hex/ascii representation of data which was outputted from the client to the child via the master file descriptor. This is only seen during tracing of the server. The `/etc/inetd.conf` must have specified `-D ptydata` or `-D all.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
utility.c

**Procedure name**
ptyflush()

EZYTU23E   UTILITY: Unexpected iconv error description rsn = errnojr

**Explanation**
This message is written to the client as well as the syslog file. The errno identified from iconv (on the EBCDIC to ascii converter) is not currently checked for. Unexpected error detected.

description describes the error.
errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
E2Amultiybyte_translate

EZYTU24E  UTILITY: Unexpected iconv error. description rsn = errnojr

Explanation
This message is written to the client as well as the syslog file. An unexpected error condition was received from the iconv call.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
E2Amultiybyte_translate

EZYTU25E  UTILITY: Unexpected iconv error description rsn = errnojr
Explanation
Building an ascii to EBCDIC translate table from the default tables. An error was produced during processing. Check the errno and errnojr for corrective action.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
initxlate()

EZYTU26E  UTILITY: Unexpected iconv error description rsn = errnojr

Explanation
Building an EBCDIC to ascii translate table from the default tables. An error was produced during processing. Check the errno and errnojr for corrective action.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
initxlate()
UTILITY: Unexpected iconv open error description rsn = errnoj

Explanation
The iconv_open failed for EBCDIC to ascii converter. The errno identified from iconv (on the EBCDIC to ascii converter) is not currently checked for. Unexpected error detected.

description describes the error.

errnoj is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
None.

System programmer response
Ensure that the converter table being specified is valid and exists.

Module
utility.c

Procedure name
doMultiByte()

UTILITY: Unexpected iconv open error description rsn = errnoj

Explanation
The iconv_open failed for ascii to EBCDIC converter. The errno identified from iconv (on the ascii to EBCDIC converter) is not currently checked for. Unexpected error detected.

description describes the error.

errnoj is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
None.

System programmer response
Ensure that the converter table being specified is valid and exists.

Module
utility.c
Procedure name
initxlate()

EYZTU29E  UTILITY: ascii translation error Xbuf->begin_translate

Explanation
The iconv failed for ascii to EBCDIC converter.

System action
Processing continues.

Operator response
None.

System programmer response
Ensure that the converter table being specified is valid and exists.

Module
utility.c

Procedure name
A2Emultibyte_translate()

EYZTU30I  UTILITY: A2E Begin_write len=bytes_translated,data=begin_w

Explanation
Tracing information inserted to help debug conversion errors. It prints out both the length of the characters being converted as well as the character set. It is invoked using -t option in the setup parameters.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
A2Emultibyte_translate()

EYZTU31I  UTILITY:E2A Begin_translate len=inlen,data=begin_w
Explanation
The iconv failed for EBCDIC to ascii converter. Tracing information inserted to help debug conversion errors. It prints out both the length of the characters being converted as well as the character set. It is invoked using -t option in the setup parameters.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
utility.c

Procedure name
E2Amultibyte_translate()
**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
None.

**Module**
utility.c

**Procedure name**
fatal()

**EZYTU34I**  
`id pri failing call code errno reason errnojr h_errno h_errno`

**Explanation**
A system or library call failed.

**id**
This identifies the location within the source code that recorded the error.

**pri**
This is the value passed as the first parameter to the syslog() function. See the description of the syslog() function in z/OS XL C/C++ Runtime Library Reference for more information.

**failing call**
This is the system or library call which failed. See the description of this system or library call in z/OS XL C/C+++ Runtime Library Reference for more information.

**errno**
This is the hexadecimal 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. If there is no errno, this field will display as **N/A**.

**errnojr**
This is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed. If there is no errnojr, this field will display as **N/A**.

**h_errno**
This error code is no longer applicable and will always display as **N/A**.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**EZYTU35I**  
Error messages will not be saved in /tmp/telnetd.stderr, reason **reason**
Explanation
Messages written to stderr in the otelnetd process are usually stored in /tmp/telnetd.stderr. These messages will be discarded for one of the following reasons:

1. /tmp/telnetd.stderr could not be opened for append or created.
2. A system or library call failed while processing /tmp/telnetd.stderr. See a prior EZYTU34I record for an explanation.
3. /tmp/telnetd.stderr exists but is not a regular file. /tmp/telnetd.stderr must be a regular file. Other types of files, such as symbolic links, cannot be used because of possible security exposures.
4. /tmp/telnetd.stderr was created again by another process while being opened and checked by this otelnetd process.

System action
Processing continues.

Operator response
None.

System programmer response
None.

EZYTU36I  code userid local_IP local_port remote_IP remote_port client_hostname

Explanation
This trace record is written to the syslogd facility auth when the -D login or -D all parameter is specified on the otelnetd command line. It provides a one-line summary of login and logout activity.

code describes which activity is being recorded:

L  The user successfully logged on to the system.
O  The user logged off.
U  The user provided a user ID which was incorrect.
P  The user provided a password which was incorrect.
C  The user did not successfully change the password.

userid is the user ID specified by the user when logging on.
local_IP is the destination IP address specified by the user when connecting to the telnet server.
local_port is the destination TCP port specified by the user when connecting to the telnet server.
remote_IP is the IP address of the host used by the client to connect to the telnet server.
remote_port is the TCP port assigned to the telnet client.
client_hostname is the DNS name of the host used by the client to connect to the telnet server, if that information is available. If not available, the client_hostname field in the trace record will contain a hyphen.
System action
Processing continues.

Operator response
None.

System programmer response
None.

**EZYTY01E**  
GETPTY: Out of ptys. **description rsn = errnojr**

Explanation
This message is written to the client as well as the syslog file. All of the /dev/ptypXXX are currently in use with other sessions or processes.

*description* describes the error.

*errnojr* is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the *z/OS UNIX System Services Messages and Codes*, where the reason codes are listed.

System action
Processing ends.

Operator response
User has to wait until some /dev/ptypXXX are freed up for use.

System programmer response
Check to determine if there are any ghosts or zombies hanging around that can be released.

Module
protocol.h

Procedure name
telnet()

**EZYTY02I**  
GETPTY: open of /dev/ptyp

Explanation
These messages are only issued if -D report or -D netdata specified in /etc/inetd.conf. They correlate to the conditions of the writing for the pty.

System action
Processing continues.

Operator response
None.

System programmer response
None.
Module
protocol.h

Procedure name
telnet()

**EZYTY03I** gotpty: ioctl TIOCSWINSIZ

Explanation
These messages are only issued if -D report or -D netdata specified in /etc/inetd.conf. They correlate to the conditions of the writing for the pty.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
sys_term.h

Procedure name
gotpty()

**EZYTY04E** GETPTY: open error on line description rsn = errnojr

Explanation
This message is written to the client as well as the syslog file. A problem occurred trying to open the corresponding /dev/ttyXXX file.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
None.

System programmer response
Correct the error indicated by description and errnojr.

- Access is denied.
- The process is busy.
• The open was interrupted by a signal.
• The system has reached the maximum number of file descriptors it can have open.
• Permission to open is denied for one of the following reasons.
  – The user who opened the master tty is not the same user associated with the slave tty.
  – Internal security error.
  – Different path name was specified for the slave than earlier opens.

Module
sys_term.h

Procedure name
gotpty()

EZYTY05I  GETPTY: slave fd = slavefd, masterfd = masterfd

Explanation
These messages are only issued if -D report or -D netdata specified in /etc/inetd.conf. They correlate to the conditions of the writing for the pty.

System action
Processing continues.

Operator response
None.

System programmer response
None.

Module
sys_term.h

Procedure name
gotpty()

EZYTY06E  gotpty: __tcsetattr description rsn = errnojr

Explanation
This message is written to the client as well as the syslog file specified in /etc/inetd.conf. They correlate to the conditions of the writing for the pty.

description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.
Operator response
None.

System programmer response
Correct the error indicated by description and errnojr.

Module
sys_term.h

Procedure name
gotpty()

EZYTY07E gotpty: __tcsetcp description rsn = errnojr

Explanation
This message is written to the client as well as the syslog file. Issued a tcsetcp to inform the tty of the code pages that we are using to translate ascii to EBCDIC and the reverse. The __tcsetcp failed. description describes the error.

errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

System action
Processing ends.

Operator response
None.

System programmer response
Correct the error indicated by description and errnojr.

Module
sys_term.h

Procedure name
gotpty()

EZYTY08I argv_fsum(argument_number) = argument_value

Explanation
These are the parameters that are passed to the spawned process that creates the child. This message is issued once for each argument. A few of these variables are set by the server.

- argv_fsum(0) = argument name
- argv_fsum(4) = master file descriptor
- argv_fsum(5) = slave file descriptor
- argv_fsum(8) = Debug tracing variable
- argv_fsum(9) = Debug tracing variable
• argv_fsum(11) = terminal type

*argument_number* is the argument number. It will be in the range 0–16.

*argument_value* is the value assigned to that argument.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
login_tty()

**EZYTY09I  login_tty: spawnp fsumocl spacing**

**Explanation**
The pid number associated with the fork() for FSUMOCLO for cleanup.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
telnetd.c

**Procedure name**
login_tty()

**EZYTY10E  login_tty: spawnp error child_pid description rsn = errno j**

**Explanation**
This message is written to the client as well as the syslog file. The pid number associated with the fork() for FSUMOCLO for cleanup.

description describes the error.
errnojr is the hexadecimal UNIX System Services reason code. The format of the 4-byte reason code is explained in the introduction to the reason codes (Errnojrs) chapter of the z/OS UNIX System Services Messages and Codes, where the reason codes are listed.

**System action**
Processing ends.

**Operator response**
None.

**System programmer response**
None.

**Module**
telnetsd.c

**Procedure name**
login_tty()

---

**EZTY11I**  GETPTY: stat of /dev/ptyp

**Explanation**
This message is only issued if -D report or -D netdata specified in /etc/inetd.conf. They correlate to the conditions of the writing for the pty.

**System action**
Processing continues.

**Operator response**
None.

**System programmer response**
None.

**Module**
sys_term.h

**Procedure name**
gotptty()

---

**EZTY12I**  ioctl masterfd TIOCEXT

**Explanation**
This message is only issued if -D report or -D netdata specified in /etc/inetd.conf. They correlate to the conditions of the writing for the pty.

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

System programmer response
None.

Module
sys_term.h

Procedure name
tty_setlinemode()

**EZYTY13E** login_tty failed.

Explanation
This message is written to the client as well as the syslog file. The login_tty() routine was called which sets up and spawns the tty. Errors were found during the processing and a return code of -1 was sent back to the caller.

System action
Processing ends.

Operator response
None.

System programmer response
Check for further messages in the log to explain reason for the failure.

Module
sys_term.h

Procedure name
gotpty()

**EZYTY14E** tcgetattr() failed on master tty

Explanation
A tcgetattr() was issued on the master tty, which is in fact an ioctl(). It returned with an unexpected error.

System action
Processing ends.

Operator response
None.

System programmer response
Correct the error indicated by description and errnojr.
Module
sys_term.h

Procedure name
init_termbuf()
Chapter 13. EZYXxxxx messages

EZYXM01  The arrow direction is not correct.
W

Explanation
XmNarrowDirection resource in XmArrowButton widget class. XmNarrowDirection resource in XmArrowButtonGadget widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ArrowB.c, ArrowBG.c

EZYXM02  Incorrect resize policy.
W

Explanation
XmNresizePolicy resource in XmBulletinBoard widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
BulletinB.c

EZYXM03  Incorrect dialog style.
W
Explanation
XmNdialogStyle resource in XmBulletinBoard widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
BulletinB.c

EZYXM04 Incorrect shadow type.

Explanation
XmNshadowType resource in XmBulletinBoard widget class. XmNshadowType resource in XmDrawnButton widget class. XmNshadowType resource in XmFrame widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
BulletinB.c

EZYXM05 Null font list (no vendor shell default).

Explanation
XmNbuttonFontList resource, XmNlabelFontList resource, or XmNtextFontList resource in XmBulletinBoard widget class.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
BulletinB.c

EZYXM06
Dialog style must be XmDIALOG_MODELESS

Explanation
The Initialize function or the SetValues function detected that the bulletin board dialog style was not set as XmDIALOG_MODELESS, when XmDIALOG_MODELESS is required.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library

Procedure name
BulletinB.c

EZYXM11
XmCascadeButton must have correct type of XmRowColumnWidgetClass parent.

Explanation
XmCascadeButton widget class. XmCascadeButtonGadget widget class.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
CascadeBG.c

EZYXM12 Only XmMENU_PULLDOWN XmRowColumnWidgets can be submenus.

Explanation
XmCascadeButton widget class. XmCascadeButtonGadget widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
CascadeB.c, CascadeBG.c

EZYXM13 MapDelay must be >= 0.

Explanation
XmNmappingDelay resource in XmCascadeButton widget class. XmNmappingDelay resource in XmCascadeButtonGadget widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
**Procedure name**
CascadeB.c, CascadeBG.c

**EZYXM14**

XmCascadeButtonGadget must have XmRowColumnWidgetClass parent with XmNrowColumnType XmMENU_PULLDOWN, XmMENU_POPUP, XmMENU_BAR or XmMENU_OPTION.

**Explanation**
XmCascadeButton widget class. XmCascadeButtonGadget widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
CascadeB.c

**EZYXM15**

XtGrabPointer failed.

**Explanation**
XGrabPointer function. XtGrabPointer function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
CascadeB.c, MenuUtil.c, TrackLoc.c

**EZYXM16**

XtGrabKeyboard failed.
Explanation
XGrabKeyboard function. XtGrabKeyboard function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MenuUtil.c, TextF.c, TextIn.c

EZYXM17  Only XmRowColumn widgets of type XmMENU_PULLDOWN can be submenus.
W

Explanation
The Initialize function or the SetValues function detected that a submenu in the processing Widget is not of type XmMENU_PULLDOWN. The submenu Widget is reset to NULL.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
CascadeB.c, CascadeBG.c

EZYXM21  The dialog type must be XmDIALOG_COMMAND.
W

Explanation
XmCommand widget class. XmNdialogType resource in XmSelectionBox widget class.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Command.c

EZYXM22
Invalid child type. The Command widget does not have this child.

Explanation
XmCommandGetChild function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Command.c

EZYXM23
Invalid XmString, check for invalid charset.

Explanation
XmCommandAppendValue function. XmCommandSetValue function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
### Module
Xm.a library.

### Procedure name
Command.c

#### EZYXM24
**Explanation**
NULL or empty string passed in to CommandAppendValue.

#### System action
The application continues.

#### System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

### Module
Xm.a library.

### Procedure name
Command.c

#### EZYXM25
**Explanation**
XmNmustMatch is always False for a Command widget.

#### System action
The application continues.

#### System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
**EZYXM26**  
XmNhistoryMaxItems must be a positive integer greater than zero.

**Explanation**  
The Initialize function or the SetValues function detected that the XmNhistoryMaxItems for the XmCommand Widget has a value less than zero. The XmNhistoryMaxItems is reset to the initialized value or 100.

**System action**  
The application continues.

**Operator response**  
Contact the system programmer.

**System programmer response**  
Use your X Window System/Motif documentation to correct this application programming error.

**Module**  
Xm.a library.

**Procedure name**  
Command.c

---

**EZYXM31**  
Must call XmClipboardStartCopy() before XmClipboardCopy().

**Explanation**  
XmClipboardCopy function.

**System action**  
The application continues.

**Operator response**  
None.

**System programmer response**  
Use your X Window System/Motif documentation to correct this application programming error.

**Module**  
Xm.a library.

**Procedure name**  
CutPaste.c

---

**EZYXM32**  
Must call XmClipboardStartCopy() before XmClipboardEndCopy().

**Explanation**  
XmClipboardEndCopy function.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
CutPaste.c

EZYXM33
Too many formats in XmClipboardCopy().

Explanation
XmClipboardCopy function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
CutPaste.c

EZYXM35
Incorrect data type.

Explanation
The ClipboardFindItem function detected that the data type passed to the Clipboard does not match the data type of the Clipboard Pointer.

System action
The application continues.

Operator response
Contact the system programmer.
**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
CutPaste.c

**EZYXM36  ClipboardCorrupt**

**Explanation**
One of the following functions detected a clipboard error, and called the ClipboardError routine with key CLIPBOARD_CORRUPT:
- ClipboardFindFormat
- ClipboardDeleteFormat
- ClipboardDeleteFormats
- ClipboardDeleteItemLabel
- ClipboardIsMarkedForDelete
- ClipboardMarkItem
- ClipboardDataIsReady
- XmClipboardEndCopy
- XmClipboardCopyByName
- XmClipboardUndoCopy
- ClipboardRetrieve

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
CutPaste.c

**EZYXM37**

**Internal error - corrupt data structure.**
**Explanation**
An internal application error was detected. One of the following functions detected that either the ClipboardDataItem or ClipboardFormatItem did not contain the required data structure:

- ClipboardFindFormat
- ClipboardDeleteFormat
- ClipboardDeleteFormats
- ClipboardDeleteItemLabel
- ClipboardIsMarkedForDelete
- ClipboardMarkItem
- ClipboardDataIsReady
- XmClipboardEndCopy
- XmClipboardCopyByName
- XmClipboardUndoCopy
- ClipboardRetrieve

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

---

**Module**
Xm.a library.

**Procedure name**
CutPaste.c

**EZYM38**
Registered format length must be 8, 16, or 32

**Explanation**
XmClipboardRegisterFormat function detected that the format_length variable did not contain an allowed value.

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
CutPaste.c

EZYXM39 Registered format name must not be NULL.

Explanation
The XmClipboardRegisterFormat function detected that the format_name variable did not contain an allowed value.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
CutPaste.c

EZYXM41 DialogShell widget only supports one rectObj child.

Explanation
XmDialogShell widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DialogS.c
gadgets aren't allowed in shell.

Explanation
XmDialogShell widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DialogS.c

DialogShell widget supports only one RectObj child

Explanation
The InsertChild function detected that the GetRectObjKid function failed because the input child is a CoreClass object, instead of the required RectObj object.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DialogS.c

Margin width or height cannot be negative.

Explanation
XmDrawingArea widget class.
**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
DrawingA.c

**EZYXM52** Incorrect resize policy.

**Explanation**
XmNresizePolicy resource in XmDrawingArea widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
DrawingA.c

**EZYXM61** Fraction base cannot be zero.

**Explanation**
XmForm widget class.

**System action**
The application continues.

**Operator response**
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Form.c

EZYXM62 Incorrect form attachment type.

Explanation
XmForm widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Form.c

EZYXM63 Cannot set constraints for non-resizable widget.

Explanation
XmForm widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Procedure name
Form.c

EZYXM64 Attachment widget must not be null.
W

Explanation
XmForm widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Form.c

EZYXM65 Circular dependency in Form children.
W

Explanation
XmForm widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Form.c

EZYXM66 Edge attached to a widget but no widget specified.
W
Explanation
XmForm widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Form.c

EZYXM67
W
Bailed out of edge synchronization after 10,000 iterations. Check for contradictory constraints on the children of this form.

Explanation
XmForm widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Form.c

EZYXM68
W
Attachment widget must be have same parent as widget.

Explanation
XmForm widget class.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Form.c

EZYXM72
W
Only one child should be inserted in a frame.

Explanation
XmFrame widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Frame.c

EZYXM73
W
Invalid margin width.

Explanation
XmFrame widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
Frame.c

EZYXM74
Invalid margin height.

Explanation
XmFrame widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Frame.c

EZYXM81
Invalid highlight thickness.

Explanation
XmGadget widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
EZYXM82  The unit type is incorrect.

**Explanation**
XmNunitType resource in XmGadget widget class. XmNunitType resource in XmManager widget class. XmNunitType resource in XmPrimitive widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Gadget.c

---

EZYXM83  Invalid shadow thickness.

**Explanation**
XmGadget widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Gadget.c

---

EZYXM84  Cannot set pixmap resource to unspecified.

**Explanation**
XmNtopShadowPixmap resource in XmPrimitive widget class. XmNbottomShadowPixmap resource in XmPrimitive widget class. XmNhighlightShadowPixmap resource in XmPrimitive widget class.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Gadget.c

EZYXM85  Cannot change XmNlayoutDirection after initialization.

Explanation
The SetValues function detected that the XmNlayoutDirection value was changed after initialization. The XmNlayoutDirection value is reset to the initialized value.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Gadget.c

EZYXM91  Invalid XmNlabelType.

Explanation
XmNlabelType resource in XmLabel widget class. XmNlabelType resource in XmLabelGadget widget class.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Label.c

EZYXM92 Invalid value in XmNalignment.

Explanation
XmNalignment resource in XmLabel widget class. XmNalignment resource in XmLabelGadget widget class. XmNstringDirection resource in XmManager widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Label.c

EZYXM93 Invalid value in XmNstringDirection.

Explanation
XmNstringDirection resource in XmLabel widget class. XmNstringDirection resource in XmLabelGadget widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
Label.c

EZYXM94
W
Invalid XmNlabelString - must be a compound string.

Explanation
XmNlabelString resource in XmLabel widget class. XmNlabelString resource in XmLabelGadget widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Label.c

EZYXM95
W
Invalid XmNacceleratorText - must be a compound string.

Explanation
XmNacceleratorText resource in XmLabel widget class. XmNacceleratorText resource in XmLabelGadget widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
List must have at least one visible item.

**Explanation**
XmNvisibleItemCount resource in XmList widget class. When changed, XmNvisibleItemCount must be at least 1.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

---

Invalid Selection Policy.

**Explanation**
XmNselectionPolicy resource in XmList widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

---

Invalid Size Policy.

**Explanation**
XmNlistSizePolicy resource in XmList widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
List.c

EZYXN03  Invalid ScrollBar Display Policy.

Explanation
XmNscrollbarDisplayPolicy resource in XmList widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
List.c

EZYXN04  Invalid String Direction.

Explanation
XmNstringDirection resource in XmList widget class.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
List.c

EZYXN05
W
Cannot change size policy after initialization.

Explanation
XmNlistSizePolicy resource in XmList widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
List.c

EZYXN06
W
Must set item count to non-negative value.

Explanation
XmNitemCount resource in XmList widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
**EZYXN07**  NULL font in SetValues ignored.

**Explanation**
XmList widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

---

**EZYXN08**  Invalid item(s) to delete.

**Explanation**
XmNItemCount resource in XmList widget class. XmListDeleteItem function. XmListDeleteItems function. XmListDeletePos function. XmListDeleteItemsPos function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

---

**EZYXN09**  No Horizontal Scrollbar to set.
**Explanation**
XmListSetHorizPos function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

---

**EZYXN10**  Invalid Margin setting.

**Explanation**
XmList widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

---

**EZYXN11**  Invalid Spacing Value.

**Explanation**
XmNlistSpacing resource in XmList widget class.

**System action**
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
List.c

EZYXN12  Cannot set items to NULL with non-zero item count.
W

Explanation
XmNitemCount resource in XmList widget class. XmNitems resource in XmList widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
List.c

EZYXN13  Must set selected item count to non-negative value.
W

Explanation
XmNselectedItemCount resource in XmList widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
List.c

EZYXN14
Cannot set selected items to NULL with non-zero item count.

Explanation
XmNselectedItemCount resource in XmList widget class. XmNselectedItems resource in XmList widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
List.c

EZYXN15
Cannot set top position less than 1.

Explanation
XmNtopItemPosition resource in XmList widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
**EZYXN16**  XmNitems and XmNitemCount mismatch!

**Explanation**
XmNitemCount resource in XmList widget class. XmNitems resource in XmList widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

---

**EZYXN17**  Cannot leave add mode in multiple selection.

**Explanation**
XmList widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

---

**EZYXN18**  XmNselectedPositionCount must not be negative.

**Explanation**
The SetValues function detected that the XmNselectedPositionCount was either negative or zero, when a positive value was required.
**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

**EZYXN19**  Cannot set XmNselectedPosition to NULL when XmNSelectedPositionCount is positive.

**Explanation**
The SetValues function detected an error with the XmNselectedPosition and XmNSelectedPositionCount. The selected position is set to NULL, but the position count contains a positive value.

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
List.c

**EZYXN21**  The Menu Bar cannot be changed to NULL.

**Explanation**
XmNmenuBar resource in XmMainWindow widget class.

**System action**
The application continues.

**Operator response**
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MainW.c

EZYXN22
W
The Command Window cannot be changed to NULL.

Explanation
XmNcommandWindow resource in XmMainWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MainW.c

EZYXN23
W
Negative margin value ignored.

Explanation
XmMainWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
**Procedure name**
MainW.c

**EZYXN31**  MenuShell widgets must have a XmRowColumnWidgetClass child.

**Explanation**
XmMenuShell widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
MenuShell.c

**EZYXN32**  Attempting to manage an incomplete menu.

**Explanation**
XmMenuShell widget class. XmCreatePopupMenu function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
MenuShell.c

**EZYXN33**  XmPopup requires a subclass of shellWidgetClass.
**Explanation**
The `_XmPopupI` function detected an error in `XtIsShell`. The passed Widget was of an incorrect class.

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
MenuShell.c

---

**EZYXN34**

W

**XmPopdown requires a subclass of shellWidgetClass.**

**Explanation**
The `_XmPopdown` function detected an error in `XtIsShell`. The passed Widget was of an incorrect class.

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
MenuShell.c

---

**EZYXN35**

W

**XtMenuPopup requires exactly one argument.**

**Explanation**
The `_XmMenuPopupAction` function detected that the argument count was not one.

**System action**
The application continues.
Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MenuShell.c

EZYXN36  XtMenuPopup only supports ButtonPress, KeyPress or EnterNotify events.
W

Explanation
The _XmMenuPopupAction function detected that the XEvent did not match the allowed values of ButtonPress, KeyPress or EnterNotify. The modal grab will be Nonexclusive.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MenuShell.c

EZYXN37  Cannot find popup widget string in XtMenuPopup.
W

Explanation
The _XmMenuPopupAction function detected a failure in the _XmFindPopup routine. The specified popup was not found.

System action
The application continues.

Operator response
Contact the system programmer.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MenuShell.c

EZYXN38 Cannot find popup widget string in XtMenuPopdown.
W

Explanation
The XmMenuPopdownAction function detected a failure in the XmFindPopup routine. The specified popup was not found.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MenuShell.c

EZYXN39 XtMenuPopdown called with more than one argument.
W

Explanation
The XmMenuPopdownAction function detected that the number of arguments was more than one.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
EZYXN40
Cannot change XmNlayoutDirection after initialization.

Explanation
The SetValues function detected that the XmNlayoutDirection value was changed after initialization. The XmNlayoutDirection value is reset to the initialized value.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MenuShell.c

EZYXN41
Invalid Dialog Type.

Explanation
XmNdialogType resource in XmMessageBox widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MessageB.c

EZYXN42
Invalid Default Button Type.
Explaination
XmNdefaultButtonType resource in XmMessageBox widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MessageB.c

EZYXN43 Invalid Alignment Type.

Explanation
XmNmessageAlignment resource in XmMessageBox widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MessageB.c

EZYXN44 Invalid Child Type.

Explanation
XmMessageBoxGetChild function.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MessageB.c

EZYXN45  PushButton Id cannot be changed directly.
W

Explanation
XmMessageBox widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
MessageB.c

EZYXN46  Use XmNdefaultButtonType to set MessageBox default button.
W

Explanation
XmMessageBox widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
MessageB.c

EZYXN54  Invalid minimum value, must be greater than zero.
Explanation
XmNpaneMinimum resource in XmPanedWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
PanedW.c

EZYXN55  Invalid maximum value, must be greater than zero.
Explanation
XmNpaneMaximum resource in XmPanedWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
EZYXN56  Invalid minimum/maximum value, minimum value must be smaller than the maximum value.

Explanation
XmNpaneMinimum resource in XmPanedWindow widget class. XmNpaneMaximum resource in XmPanedWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
PanedW.c

EZYXN57  Constraints do not allow appropriate sizing.

Explanation
XmPanedWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
PanedW.c

EZYXN58  Too few parameters.

Explanation
XmPanedWindow widget class.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
PanedW.c

EZYXN59  Invalid 1st parameter.
W

Explanation
XmPanedWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
PanedW.c

EZYXN61  fontList is not defined.
W

Explanation
XmPanedWindow widget class.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
PanedW.c

EZYXN71 Must be a vendor shell.

Explanation
XmAddProtocols function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Protocols.c

EZYXN72 Protocol manager already exists.

Explanation
XmAddProtocols function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
More protocols than I can handle.

Explanation
XmAddProtocols function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Not enough memory.

Explanation
There was not enough memory to perform the function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Attempt to set width to zero. Set to default value 16.
Explanation
XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYX002 Attempt to set width to zero. The value is ignored.

Explanation
XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYX003 Attempt to set height to zero. Set to default value 16.

Explanation
XmRowColumn widget class.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYX004 W
Attempt to set height to zero. The value is ignored.

Explanation
XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYX005 W
XmNhelpWidget not used by PopUps. It is set to NULL.

Explanation
XmNmenuHelpWidget resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO06  XmNhelpWidget not used by Pulldowns. It is set to NULL.
W

Explanation
XmNmenuHelpWidget resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO07  XmNhelpWidget not used by Option menus. It is set to NULL.
W

Explanation
XmNmenuHelpWidget resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
EZYXO08  XmNhelpWidget not used by Work Areas. It is set to NULL.

Explanation
XmNmenuHelpWidget resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

---

EZYXO09  Unknown value of XmNrowColumnType. It is set to WorkArea.

Explanation
XmNrowColumnType resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

---

EZYXO10  Widget hierarchy not appropriate for this XmNrowColumnType. It is set to WorkArea.

Explanation
XmCreatePulldownMenu function.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO11 Attempt to change XmNrowColumnType after initialization. The value is ignored.

Explanation
XmNrowColumnType resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO12 Unknown value of XmNorientation. The default value is used.

Explanation
XmNorientation resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO13 Attempt to set XmNorientation to unknown value. The value is ignored.

Explanation
XmNorientation resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO14 Unknown value of XmNpacking. The default value is used.

Explanation
XmNpacking resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
**Procedure name**
RowColumn.c

**EZYX015** Attempt to set XmNpacking to unknown value. The value is ignored.

**Explanation**
XmNpacking resource in XmRowColumn widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
RowColumn.c

**EZYX016** Unknown value of XmNentryAlignment. The default value is used.

**Explanation**
XmNentryAlignment resource in XmRowColumn widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
RowColumn.c

**EZYX017** Attempt to set XmNentryAlignment to unknown value. The value is ignored.
**Explanation**
XmNentryAlignment resource in XmRowColumn widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
RowColumn.c

**EZYXO18**
Attempt to set XmNisHomogenous to FALSE for a RowColumn widget of type XmMENU_BAR. The value is ignored.

**Explanation**
XmNisHomogeneous resource in XmRowColumn widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
RowColumn.c

**EZYXO19**
Attempt to change XmNentryClass for a RowColumn widget of type XmMENU_BAR. The value is ignored.

**Explanation**
XmNentryClass resource in XmRowColumn widget class.

**System action**
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO20 Attempt to change XmNwhichButton via XtSetValues for a RowColumn widget of type XmMENU_PULLDOWN. The value is ignored.

Explanation
XmNwhichButton resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO21 Attempt to change XmNmenuPost via XtSetValues for a RowColumn widget of type XmMENU_PULLDOWN. The value is ignored.

Explanation
XmNmenuPost resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO22

Attempt to set XmNpostMenu to an illegal value. The value is ignored.

Explanation
XmNmenuPost resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO23

Attempt to change XmNshadowThickness for a RowColumn widget not of type XmMENU_PULLDOWN or XmMENU_POPUP. The value is ignored.

Explanation
XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
EZYXO24  Attempt to change XmNorientation for a RowColumn widget of type XmMENU_OPTION. The value is ignored.

Explanation
XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO25  Attempt to add wrong type child to a menu (i.e. RowColumn) widget

Explanation
XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

EZYXO26  Attempt to add wrong type child to a homogeneous RowColumn widget.

Explanation
XmNisHomogeneous resource in XmRowColumn widget class.
**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
RowColumn.c

**EZYXO27**
Attempt to change XmNisHomogeneous for a RowColumn widget of type XmMENU_OPTION ignored.

**Explanation**
XmNisHomogeneous resource in XmRowColumn widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
RowColumn.c

**EZYXO28**
Tear off enabled on a shared menupane is allowed but not recommended.

**Explanation**
Tear-off Menus in XmRowColumn widget class.

**System action**
The application continues.

**Operator response**
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

**EZYXO29** Illegal mnemonic character. Could not convert X KEYSYM to a keycode.

Explanation
XmNmnemonic resource in XmRowColumn widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RowColumn.c

**EZYXO31** The scale minimum value is greater than or equal to the scale maximum value.

Explanation
XmNminimum resource in XmScale widget class. XmNmaximum resource in XmScale widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
**Procedure name**
Scale.c

**EZYXO32** — The specified scale value is less than the minimum scale value.

**Explanation**
XmNvalue resource in XmScale widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Scale.c

**EZYXO33** — The specified scale value is greater than the maximum scale value.

**Explanation**
XmNvalue resource in XmScale widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Scale.c

**EZYXO34** — Incorrect orientation.
Explanation
XmNorientation resource in XmScale widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Scale.c

<table>
<thead>
<tr>
<th>EZYXO35</th>
<th>Incorrect processing direction.</th>
</tr>
</thead>
</table>

Explanation
XmNprocessingDirection resource in XmScale widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Scale.c

<table>
<thead>
<tr>
<th>EZYXO36</th>
<th>Invalid highlight thickness.</th>
</tr>
</thead>
</table>

Explanation
XmNhighlightThickness resource in XmScale widget class.

System action
The application continues.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Scale.c

EZYX037 Invalid scaleMultiple; greater than (max - min).

Explanation
XmNscaleMultiple resource in XmScale widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Scale.c

EZYX038 Invalid scaleMultiple; less than zero.

Explanation
XmNscaleMultiple resource in XmScale widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
Scale.c

EZYXO39 (Maximum - minimum) cannot be greater than INT_MAX divided by 2. Minimum has been set
to zero. Maximum may have been set to (INT_MAX/2).

Explanation
XmNmaximum resource in XmScale widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Scale.c

EZYXO41 The scrollbar minimum value is greater than or equal to the scrollbar maximum value.

Explanation
XmNminimum resource in XmScrollBar widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
**EZYXO42**
The specified slider size is less than 1.

**Explanation**
XmNincrement resource in XmScrollBar widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ScrollBar.c

---

**EZYXO43**
The specified scrollbar value is less than the minimum scrollbar value.

**Explanation**
XmNvalue resource in XmScrollBar widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ScrollBar.c

---

**EZYXO44**
The specified scrollbar value is greater than the maximum scrollbar value minus the scrollbar slider size.

**Explanation**
XmNvalue resource in XmScrollBar widget class.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollBar.c

**EZYXO45**  Incorrect orientation.

Explanation
XmNorientation resource in XmScrollBar widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollBar.c

**EZYXO46**  Incorrect processing direction.

Explanation
XmNprocessingDirection resource in XmScrollBar widget class.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollBar.c

EZYXO47 The scrollbar increment is less than 1.

Explanation
XmNincrement resource in XmScrollBar widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollBar.c

EZYXO48 The scrollbar page increment is less than 1.

Explanation
XmNpageIncrement resource in XmScrollBar widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
Procedure name
ScrollBar.c

EZYXO49
The scrollbar initial delay is less than 1.

Explanation
XmNinitialDelay resource in XmScrollBar widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollBar.c

EZYXO50
The scrollbar repeat delay is less than 1.

Explanation
XmNrepeatDelay resource in XmScrollBar widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollBar.c

EZYXO51
Error in context manager; scrollbar backgrounds cannot be set correctly
Explanation
XmScrollBar widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollBar.c

EZYXO52  Error in context manager; scrollbar foregrounds cannot be set correctly.

Explanation
XmScrollBar widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollBar.c

EZYXO53  Specified slider size is greater than the scrollbar maximum value minus the scrollbar minimum value.

Explanation
XmNsliderSize resource in XmScrollBar widget class.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollBar.c

EZYXO61
Invalid ScrollBar Display policy.

Explanation
XmNscrollBarDisplayPolicy resource in XmScrolledWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrolledW.c

EZYXO62
Invalid Scrolling Policy.

Explanation
XmNscrollingPolicy resource in XmScrolledWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
**Module**
Xm.a library.

**Procedure name**
ScrolledW.c

---

**EZYXO63**

**Invalid Visual Policy.**

**Explanation**
XmNvisualPolicy resource in XmScrolledWindow widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

---

**Module**
Xm.a library.

**Procedure name**
ScrolledW.c

---

**EZYXO64**

**Invalid placement policy.**

**Explanation**
XmNscrollBarPlacement resource in XmScrolledWindow widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.
EZYX066 Cannot change scrolling policy after initialization.

Explanation
XmNscrollingPolicy resource in XmScrolledWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrolledW.c

EZYX067 Cannot change visual policy after initialization.

Explanation
XmNvisualPolicy resource in XmScrolledWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrolledW.c

EZYX068 Cannot set AS_NEEDED scrollbar policy with a visual policy of VARIABLE.

Explanation
XmNvisualPolicy resource in XmScrolledWindow widget class.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrolledW.c

EZYXO69  Cannot change scrollbar widget in AUTOMATIC mode.
W

Explanation
XmNscrollingPolicy resource in XmScrolledWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrolledW.c

EZYXO70  Cannot change clip window.
W

Explanation
XmNclipWindow resource in XmScrolledWindow widget class.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrolledW.c

**EZYX071** Cannot set visual policy of CONSTANT in APPLICATION_DEFINED mode.

Explaination
XmNvisualPolicy resource in XmScrolledWindow widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrolledW.c

**EZYX072** Wrong parameters passed to the function.

Explaination
XmScrollVisible function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
Incorrect dialog type.

Explanation
XmNdialogType resource in XmSelectionBox widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Dialog type cannot be modified.

Explanation
XmNdialogType resource in XmSelectionBox widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Only one work area child allowed.
Explanation
XmSelectionBox widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
SelectioB.c

EZYXO84
Invalid child type.

Explanation
XmSelectionBoxGetChild function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
SelectioB.c

EZYXO91
Invalid separator type.

Explanation
XmNseparatorType resource in XmSeparator widget class. XmNseparatorType resource in XmSeparatorGadget widget class.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Separator.c

EZYX092 Invalid orientation.

Explanation
XmNorientation resource in XmSeparator widget class. XmNorientation resource in XmSeparatorGadget widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Separator.c

EZYXP01 Invalid source, source ignored.

Explanation
XmNsSource resource in XmText widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
Text.c

EZYXP02 Invalid edit mode.

Explanation
XmNeditMode resource in XmText widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Text.c

EZYXP03 Text widget is editable; traversalOn must be true.

Explanation
XmNeditable resource in XmText widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Text.c
EZYXP04  Can't find position in MovePreviousLine().

Explanation
XmText widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
TextIn.c

EZYXP05  Invalid rows, must be > 0.

Explanation
XmNrows resource in XmText widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
TextOut.c

EZYXP06  XmFontListInitFontContext failed.

Explanation
XmNfontList resource in XmText widget class.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
TextF.c, TextOut.c

EZYXP07  XmFontListGetNextFont failed.
W

Explanation
XmNfontList resource in XmText widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
TextF.c, TextOut.c

EZYXP08  Character char is not supported in font. It is discarded.
W

Explanation
XmNfontList resource in XmText widget class.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
TextF.c

EZYXP09  String string is not supported in font. It is discarded.

Explanation
XmNfontList resource in XmText widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
TextF.c

EZYXP10  Cannot use a multibyte locale without a fontset. The value is discarded.

Explanation
XmNfontList resource in XmText widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Procedure name
TextF.c

**EZYXP11**  Invalid cursor position, must be >= 0.

**Explanation**
XmNcursorPosition resource in XmTextField widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

---

**Procedure name**
TextF.c

**EZYXP12**  Invalid columns, must be > 0.

**Explanation**
XmNcolumns resource in XmText widget class. XmNcolumns resource in XmTextField widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

---

**Procedure name**
TextF.c, TextOut.c

**EZYXP13**  XmNtraversalOn must always be true.

**Explanation**

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

---
Explanation
XmTextField widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
TextF.c

EZYXP14
Invalid columns, must be >= 0.

Explanation
XmNcolumns resource in XmTextField widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
TextF.c

EZYXP21
Indicator type should be either XmONE_OF_MANY or XmN_OF_MANY.

Explanation
XmNindicatorType resource in XmToggleButton widget class. XmNindicatorType resource in XmToggleButtonGadget widget class.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ToggleB.c

EZYXP31  Invalid value for navigation_type.

Explanation
XmNnavigationType resource in XmManager widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Traversal.c

EZYXP32  Wrong value in old for navigation_type!!

Explanation
XmNnavigationType resource in XmManager widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
Traversal.c

EZYXP33  Traversal bootstrap situation with bad parameters.

Explanation
XmManager widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Traversal.c

EZYXP34  Attempt to traverse to new tab using bad parameters.

Explanation
XmManager widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
**EZYXP35**  
startWidget is not in child list.

**Explanation**  
XmManager widget class.

**System action**  
The application continues.

**Operator response**  
None.

**System programmer response**  
Use your X Window System/Motif documentation to correct this application programming error.

**Module**  
Xm.a library.

**Procedure name**  
Traversal.c

---

**EZYXP36**  
Bad parameters to TraverseToChild.

**Explanation**  
XmManager widget class.

**System action**  
The application continues.

**Operator response**  
None.

**System programmer response**  
Use your X Window System/Motif documentation to correct this application programming error.

**Module**  
Xm.a library.

**Procedure name**  
Traversal.c

---

**EZYXP41**  
Invalid value for delete response.

**Explanation**  
XmNdeleteResponse resource in VendorShell widget class.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VendorS.c

EZYXP42
W
Invalid XmNpreeditType, default to OverTheSpot.

Explanation
XmNpreeditType resource in VendorShell widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VendorS.c

EZYXP43
W
Invalid value for XmNinputPolicy.

Explanation
The VendorExtSetValues function detected an incorrect value for the XmVendorShellExtPartPtr input policy. The policy is required to be either XmPER_SHELL or XmPER_WIDGET.

System action
The application continues.

Operator response
Contact the system programmer.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VendorS.c

EZYXP44  XmNlayoutDirection cannot be changed.
W

Explanation
The VendorExtSetValues function detected that the XmNlayoutDirection value was changed after initialization. The XmNlayoutDirection value is reset to the initialized value.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VendorS.c

EZYXP45  FetchUnitType: bad widget class.
W

Explanation
VendorShell widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
EZYXP46 String to no-op conversion needs no extra arguments.

Explanation
VendorShell widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VendorSE.c

EZYXP47 FetchUnitType called without a widget to reference.

Explanation
VendorShell widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VendorSE.c

EZYXP48E Fatal Error: _XmGetDefaultDisplay cannot be used prior to VendorS.Initialize, returns NULL.

Explanation
_XmDisplayHandle was not set before the _XmGetDefaultDisplay function was called. A NULL value is returned.
System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VendorS.c

EZYXP51 Virtual bindings Initialize hasn't been called.

Explanation
XmVirtKeys widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VirtKeys.c

EZYXP61 Invalid color requested from _XmAccessColorData.

Explanation
XmGetColors function.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Visual.c

EZYX62
Cannot allocate colormap entry for background, setting background to white.

Explanation
XmNbackground resource in Core widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Visual.c

EZYX63
Cannot parse given background color, setting background to white.

Explanation
XmNbackground resource in Core widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
**Procedure name**  
Visual.c

**EZYXP71**  
The specified Input Method failed to init : string.

**Explanation**  
XmNinputMethod resource in VendorShell widget class.

**System action**  
The application continues.

**Operator response**  
None.

**System programmer response**  
Use your X Window System/Motif documentation to correct this application programming error.

**Module**  
Xm.a library.

**Procedure name**  
XmIm.c

**EZYXP72**  
Cannot create the Input Method Object.

**Explanation**  
XmNinputMethod resource in VendorShell widget class.

**System action**  
The application continues.

**Operator response**  
None.

**System programmer response**  
Use your X Window System/Motif documentation to correct this application programming error.

**Module**  
Xm.a library.

**Procedure name**  
XmIm.c

**EZYXP73**  
XmIMFocus invoked with NULL widget.

**Explanation**  

**System action**  
The application continues.

**Operator response**  
None.

**System programmer response**  
Use your X Window System/Motif documentation to correct this application programming error.

**Module**  
Xm.a library.
Explanation
XmNinputMethod resource in VendorShell widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
XmIm.c

EZYXP74
XmIMMove invoked without the Input Method focus.

Explanation
XmImGetXIM function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
XmIm.c

EZYXP81
No context found for extension.

Explanation
When trying to map the extension to a context, no context was found.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
BaseClass.c

EZYXP82 _XmPopWidgetExtData: no extension found with XFindContext.

Explanation
This is a debug message.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
BaseClass.c

EZYXP83 XmFreeWidgetExtData is an unsupported routine.

Explanation
This routine is no longer a supported interface.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
**Module**

Xm.a library.

**Procedure name**

BaseClass.c

**EZYXP84**

getLabelSecResData: not enough memory.

**Explanation**

There was not enough memory to perform the requested function.

**System action**

The application continues.

**Operator response**

None.

**System programmer response**

Use your X Window System/Motif documentation to correct this application programming error.

---

**Module**

Xm.a library.

**Procedure name**

BaseClass.c

**EZYXP91**

Creating multiple XmDisplays for the same X display. Only the first XmDisplay created for a particular X display can be referenced by calls to XmGetXmDisplay.

**Explanation**

XmDisplay widget class.

**System action**

The application continues.

**Operator response**

None.

**System programmer response**

Use your X Window System/Motif documentation to correct this application programming error.

---

**Module**

Xm.a library.

**Procedure name**

Display.c
**EZYXP92**  Received TOP_LEVEL_LEAVE with no active DragContext.

**Explanation**
Function requested TOP_LEVEL_LEAVE but there was no DragContext found.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Display.c

---

**EZYXP93**  Cannot set XmDisplay class to a non-subclass of XmDisplay.

**Explanation**
XmDisplayClass widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Display.c

---

**EZYXQ01**  The _MOTIF_DRAG_WINDOW has been destroyed.

**Explanation**
The window is no longer a valid context.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragBS.c

EZYXQ02 The protocol version levels do not match.

Explanation
The level of the Motif server and client code drag protocol is not the same.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragBS.c, DragICC.c

EZYXQ03 Unable to open display.

Explanation
XmInternAtom function.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragBS.c

EZYXQ04
The atoms table is empty.

Explanation
WriteAtomsTable function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragBS.c

EZYXQ05
The target table is empty.

Explanation
WriteTargetsTable function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
The target table has an inconsistent property.

**Explanation**
ReadTargetsTable function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

---

Invalid target table index.

**Explanation**
ReadTargetsTable function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

---

GenerateCallback does not expect XmCR_DROP_SITE_ENTR as a reason.

**Explanation**

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.
Explanation
GenerateClientCallback function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragC.c

EZYXQ12 Invalid selection in DropConvertCallback.

Explanation
DropConvertCallback function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragC.c

EZYXQ13 The drop selection was lost.

Explanation
DropLoseSelection function.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragC.c

**EZYXQ14**
W

XGrabPointer failed.

Explanation
Call to lower level function, XGrabPointer, was unsuccessful.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragC.c

**EZYXQ15**
W

ExternalNotifyHandler: the callback reason is not acceptable.

Explanation
XmDragStart function. The drag message did not contain a defined reason code.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
DragC.c

**EZYXQ16**
XMDragStart must be called as a result of a button press.

**Explanation**
XMDragStart function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragC.c

**EZYXQ21**
Unknown drag and drop message type.

**Explanation**
Message type received was not expected in this context.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragICC.c
The protocol version levels do not match.

**Explanation**
The `XmGetDragReceiverInfo` function detected that the `xmDragReceiverInfoStruct` protocol version did not match the `_MOTIF_DRAG_PROTOCOL_VERSION`.

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
DragICC.c

---

No geometry specified for dragIcon pixmap.

**Explanation**
`XmDragIcon` widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
DragIcon.c

---

dragIcon created with no pixmap.

**Explanation**
`XmNPixmap` resource in `XmDragIcon` widget class.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragIcon.c

EZYXQ33
String to Bitmap converter needs Screen argument.

Explanation
XmCvtStringToBitmap function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragIcon.c

EZYXQ41
Depth mismatch.

Explanation
XmNblendModel resource in XmDragContext widget class.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragOverS.c

EZYXQ42 Unknown icon attachment.
W

Explanation
XmNattachment resource in XmDragIcon widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragOverS.c

EZYXQ43 Unknown drag state.
W

Explanation
GetDragIconColors function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
**Procedure name**
DragOverS.c

**EZYXQ44**
Unknown blendModel.

**Explanation**
XmNblendModel resource in XmDragContext widget class.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
DragOverS.c

**EZYXQ51**
Unable to get dropSite window geometry.

**Explanation**
CreateAnimationSaveData function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
DragUnder.c

**EZYXQ52**
Invalid animationPixmapDepth.
Explanation
XmNanimationPixmapDepth resource in XmDropSite registry.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DragUnder.c

EZYXQ61 Cannot create drop sites which are children of a simple drop site.
W

Explanation
XmNdropSiteType resource in XmDropSite registry.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DropSMgr.c

EZYXQ62 Receiving Motion Events without an active drag context.
W

Explanation
XmDragContext widget class.

System action
The application continues.
**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
DropSMgr.c

---

EZYXQ63  Receiving operation changed without an active drag context.

**Explanation**
XmDragContext widget class.

**System action**
The application continues.

---

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
DropSMgr.c

---

EZYXQ64  Creating an active drop site with no drop procedure.

**Explanation**
XmDropSiteRegister function.

**System action**
The application continues.

---

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.
Cannot set rectangles or rectangle numbers of composite drop sites.

**Explanation**
XmNdropRectangles resource in XmDropSite registry.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

Registering a widget as a drop site out of sequence. Ancestors must be registered before any of their descendants are registered.

**Explanation**
XmDropSiteRegister function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.
EZYXQ67 Cannot register widget as a drop site more than once.

Explanation
XmDropSiteRegister function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DropSMgr.c

EZYXQ68 DropSite type may only be set at creation time.

Explanation
XmNdropSiteType resource in XmDropSite registry.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DropSMgr.c

EZYXQ69 Cannot change rectangles of non-simple drop site.

Explanation
XmNdropRectangles resource in XmDropSite registry.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DropSMgr.c

**EZYXQ70**

Cannot register a drop site which is a descendent of a simple drop site.

Explanation
XmDropSiteRegister function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DropSMgrI.c

**EZYXQ71**

Cannot create a discontiguous child list for a composite drop site.

Explanation
XmDropSiteRegister function.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DropSMgrI.c

EZYXQ72  string is not a drop site child of string.
W

Explanation
XmDropSiteRegister function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
DropSMgrI.c

EZYXQ73  Cannot register a Shell as a drop site.
W

Explanation
The XmDropSiteRegister function detected an error while attempting to register a drop site. The Widget to be registered is defined as a Shell.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error. error.

Module
Xm.a library.
Procedure name
DropSMgrI.c

EZYXQ81 Failure of geometry request to "almost" reply.

Explanation
The Intrinsics protocol guarantees a "Yes" response to a request with identical geometry to that which was returned by a previous request returning "almost".

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
GeoUtils.c

EZYXQ82 Invalid order found in XmSelectionBox.

Explanation
XmSelectionBox geometry.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
GeoUtils.c

EZYXQ91 Memory error.
Explanation
An attempt to allocate memory failed.

System action
The application continues.

Operator response
None.

System programmer response
Increase the region size in which the program executes.

Module
Xm.a library.

Procedure name
Region.c

EZYXR01 Illegal representation type id.

Explanation
XmRepTypeValidValue function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RepType.c

EZYXR02 Illegal value number for rep type XmRxxx.

Explanation
XmRepTypeValidValue function.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RepType.c

EZYXR03 Reverse Conversion of
W

Explanation
This is a header message for XmRepTypeEntry conversions in the ReverseConvertRepType function, and will be output for most conversions.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
RepType.c

EZYXR11 FetchUnitType: bad widget class.
W

Explanation
FetchUnitType function. Widget is not gadget, manager, or primitive.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
ResConvert.c

EZYXR12
Cannot continue because of errors in a default font list.

Explanation
GetNextFontListEntry function failed.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ResConvert.c

EZYXR13
Missing colon in font string string; any remaining fonts in list unparsed.

Explanation
The GetNextFontListEntry function detected that a font set in the font list was missing the colon delimiter. Any fonts remaining on the list are not parsed.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Invalid delimiter in tag string; any remaining fonts in list unparsed.

Explanation
The GetNextFontListEntry function detected that the delimiter between two font list items is not valid. Any fonts remaining in the list are not parsed.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ResConvert.c

Unmatched quotation marks in tag string; any remaining fonts in list unparsed.

Explanation
The GetFontName function detected that the font list started with a quotation mark ("), but did not end with a quotation mark ("). Remaining fonts in the list are not parsed.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ResConvert.c

Null tag found when converting to type string; any remaining fonts in list unparsed.
**Explanation**
The GetFontTag function found a NULL tag when expecting a real value. Remaining fonts in the list are not parsed.

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

---

**Module**
Xm.a library.

**Procedure name**
ResConvert.c

**EZYXR17**
Cannot convert XmString to Compound Text.

---

**Explanation**
The XmCvtXmStringToCT function detected an error in the cvtXmStringToText function. The XmString could not be converted directly to a Compound String.

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

---

**Module**
Xm.a library.

**Procedure name**
ResEncod.c

**EZYXR18**
Insufficient memory for XmbTextListToTextProperty.

---

**Explanation**
The processCharsetAndText function detected that the XNoMemory flag was turned on during XmbTextListToTextProperty processing.
System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ResEncod.c

EZYXR19 W
Locale not supported for XmbTextListToTextProperty.

Explanation
The processCharsetAndText function detected that the XLocaleNotSupported flag was turned on during XmbTextListToTextProperty processing.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ResEncod.c

EZYXR20 W
XmbTextListToTextProperty failed.

Explanation
The processCharsetAndText function detected a failure in the XmbTextListToTextProperty function.

System action
The application continues.

Operator response
Contact the system programmer.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ResEncod.c

EZYXR21  Icon screen mismatch.

Explanation
XmScreen widget class.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Screen.c

EZYXR22  Cannot get XmScreen because XmDisplay was not found.

Explanation
XmGetXmScreen function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
**Procedure name**
Screen.c

**EZYXR41**
Could not allocate memory for color object data.

**Explanation**
An attempt to allocate memory failed.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Increase the region size in which the program executes.

**Module**
Xm.a library.

**Procedure name**
ColorObj.c

**EZYXR42**
Bad screen number from color server selection.

**Explanation**
GetSelection function.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ColorObj.c CDE Icon Gadget messages

**EZYXR51**
Cannot convert widget name to Widget.
**Explanation**
The StringToEntity function detected that an incorrect number of parameters was used.

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ResConvert.c

---

**EZYXR52**
Cannot convert compound text to XmString.

**Explanation**
The XmCvtTextToXmString function detected an error while converting a compound text string to an XmString.

**System action**
The application continues.

**Operator response**
Contact the system programmer.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ResEncod.c

---

**EZYXR53**
Cannot convert XmString to compound text.

**Explanation**
The XmCvtXmStringToText function detected an error while converting the XmString to an ASCII string.

**System action**
The application continues.
Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ResEncod.c

**EZYXR61** Applications cannot add children to XmComboBox widgets.

Explanation
The InsertChild function detected that the application attempted to add another child after the ComboBox had already made its children.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ComboBox.c

**EZYXR62** XmNcomboBoxType resource cannot be changed by XtSetValues.

Explanation
The SetValues function detected that the ComboBox type was illegally changed after creation. The ComboBox type is reset to the original value.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ComboBox.c

EZYXR63 Internal widget has been destroyed. Behavior is undefined.

Explanation
The ListSelectionCB function or the DoLayout function detected that the Widget passed into the function is missing or not valid.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ComboBox.c

EZYXR64 Internal widget has been unmanaged. Behavior is undefined.

Explanation
The DoLayout function detected that either the List or EditBox XmComboBoxWidget is unmanaged.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
procedure name
ComboBox.c

EZYXR65  XmQUICK_NAVIGATE is only valid for ComboBoxes of XmNcomboBoxType XmDROP_DOWN_LIST.

Explanation
The Initialize function or the SetValues function detected that XmQUICK_NAVIGATE was not valid with XmNcomboBoxType XmDROP_DOWN_LIST. The XmNmatchBehavior is reset to its previous state.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ComboBox.c

EZYXR66  Action invoked with the wrong number of parameters.

Explanation
The CBListAction function detected that an incorrect number of parameters was used. The required number of parameters is one.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ComboBox.c

EZYXR67  Action routine called from a widget that is not a descendant of ComboBox.

968  z/OS Communications Server: IP Messages: Volume 3 (EZY)
**Explanation**
The FindComboBox call failed in one of the following functions because the passed Widget did not contain a ComboBox:
- CBArmAndDropDownList
- CBDisarm
- CBDropDownList
- CBFocusIn
- CBFocusOut
- CBTextFocusOut
- CBArmActivate, CBCanCancel
- CBListAction

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ComboBox.c

**EZYXR68**
XmComboBoxSelectItem called with an item not present in the ComboBox.

**Explanation**
The XmComboBoxSelectItem function detected that the selected item is not present in the ComboBox.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.
**Procedure name**
ComboBox.c

**EZYXR69**  
XmComboBoxSetItem called with an item present in the ComboBox.

**Explanation**
The XmComboBoxSetItem function detected a failure in the XmListItemPos function due to an incorrect XmString value being passed into either XmComboBoxSetItem or XmListItemPos.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ComboBox.c

---

**EZYXR70**  
XmComboBoxDeletePos called with an invalid position.

**Explanation**
The XmComboBoxDeletePos function detected one of the following:
- The position to be deleted had a value lower than zero.
- The position to be deleted was greater than the number of items in the list.
- The number of list items was zero.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ComboBox.c
**EZYXR71**  XmComboBox utility routine called with an invalid widget.

**Explanation**
One of the following functions detected that passed Widget is not a ComboBox:
- XmComboBoxAddItem
- XmComboBoxDeletePos
- XmComboBoxSelectItem
- XmComboBoxSetItem
- XmComboBoxUpdate

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ComboBox.c

**EZYXR72**  Applications may not set the automatic XmComboBox widget children.

**Explanation**
The Initialize function or the SetValues function detected that a change in either the list child or the editbox child occurred. The children are reset to NULL.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
ComboBox.c
EZYXR81  Action invoked with the wrong number of parameters.

Explanation
One of the following functions was invoked with the wrong number of parameters:

- ContainerHandleBtn1Down
- ContainerHandleBtn1Motion
- ContainerHandleBtn1Up
- ContainerHandleBtn2Down
- ContainerHandleBtn2Motion

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Container.c

EZYXR82  XmNdetailColumnHeading and XmNdetailColumnHeadingCount do not match!

Explanation
The Initialize function detected that the XmNdetailColumnHeading and XmNdetailColumnHeadingCount do not match.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Container.c
**EZYXR91**  Widget class *string* has invalid CompositeClassExtension record.

**Explanation**
The FindCompClassExtension function detected that the current Widget has incorrect version and record_size values in the CompositeClassExtension record.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Manager.c

---

**EZYXR92**  Cannot change XmNlayoutDirection or XmNstringDirection after initialization.

**Explanation**
The SetValues function detected that either the XmNlayoutDirection or XmNstringDirection was changed after initialization. The incorrect value is reset to its initialized value.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Manager.c

---

**EZYXS01**  XmNnotebookChildType resource cannot be set by XtSetValues.

**Explanation**

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Manager.c
Explanation
The ConstraintSetValues function detected that the XmNnotebookChildType does not match the initialized value. The XmNnotebookChildType is reset to its original value.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Notebook.c

EZYXS11  Wrong number of parameters for CvtStringToIconPixmap.

Explanation
The CvtStringToIconPixmap function detected that an incorrect number of parameters was used.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
PixConv.c

EZYXS21  Cannot change XmNlayoutDirection after initialization.

Explanation
The SetValues function detected that the XmNlayoutDirection value was changed after initialization. The XmNlayoutDirection value is reset to its initialized value.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Primitive.c

EZYXS31  AssocNavigator requires a navigator trait.

Explanation
The _XmSFAddNavigator function detected that the XmNavigatorTrait value was set to NULL.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollFrameT.c

EZYXS32  DeAssocNavigator requires a navigator trait.

Explanation
The _XmSFRemoveNavigator function detected that the XmNavigatorTrait value was set to NULL.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
ScrollFrameT.c

EZYXS41  No items supplied for XmSTRING child.
W

Explanation
The ConstraintInitialize function or the ConstraintSetValues function detected that there were no items placed in the XmStringTable, but there were items passed to the function.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
SpinB.c

EZYXS42  XmNincrementValue cannot be 0. A value of 1 will be used.
W

Explanation
The ConstraintInitialize function or the ConstraintSetValues function detected an incorrect XmNincrementValue. XmNincrementValue is reset to one.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
SpinB.c

**EZYXS43** Spin direction specified by XmNincrementValue has been reversed to match the specified XmNminimumValue and XmNmaximumValue.

**Explanation**
The ConstraintInitialize function or the ConstraintSetValue function detected that the Spin direction was incorrect. The spin direction was reversed.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
SpinB.c

**EZYXS44** XmNposition out of range. Minimum XmNposition used.

**Explanation**
The ValidatePositionValue function detected that XmNposition was lower than the minimum allowed value. XmNposition is reset to the minimum value.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.
Procedure name
SpinB.c

EZYXS45  XmNposition out of range. Maximum XmNposition used.
W

Explanation
The ValidatePositionValue function detected that XmNposition was greater than the maximum allowed value. XmNposition is reset to the maximum value.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
SpinB.c

EZYXS46  Invalid value for XmNpositionType. Using default value.
W

Explanation
The ConstraintInitialize or the ConstraintSetValues function detected that the XmNpositionType value is incorrect.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
SpinB.c

EZYXS51  Calling SelectionCallbackWrapper when transfers should be finished
W
**Explanation**
The SelectionCallbackWrapper function was called when all outstanding transfers had completed.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Transfer.c

---

**EZYXS52**

The format and type of the callback supplied data does not match the data being merged.

**Explanation**
The XmeConvertMerge function detected that the format and type of the callback supplied data does not match the format and type of the data to be merged.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Transfer.c

---

**EZYXS53**

The status in the XmConvertCallbackStruct is not XmCONVERT_MERGE.

**Explanation**
The XmeConvertMerge function detected that the XmCONVERT_MERGE was not set, but it is required.

**System action**
The application continues.
**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Transfer.c

---

**EZYXS54**

**Explanation**
The _XmConvertHandler function detected that XmCONVERT_MORE is being used, which is not supported by this release of the Motif specification. The flag is changed to XmCONVERT_DEFAULT.

**System action**
The application continues.

---

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
Transfer.c

---

**EZYXS55**

**Explanation**
The GetSafeAtomName function detected that XGetAtomName returned with either NULL or an incorrect atom name.

**System action**
The application continues.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Transfer.c

EZYXS56 Warning: Attempt to start a MULTIPLE transfer when one is in progress.
W

Explanation
The XmTransferStartRequest function detected a MULTIPLE request while processing the current transfer.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
Transfer.c

EZYXS57 Warning: Attempt to send a MULTIPLE transfer when one is not in progress.
W

Explanation
The XmTransferSendRequest function detected that the attempted MULTIPLE request is not allowed for this transfer.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
EZYXS61  XtVaTypedArg conversion needs non-null widget handle.
Explanation
The Widget passed to the _XmTypedArgToArg function was NULL.
System action
The application continues.
Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
VaSimple.c
EZYXS62  Unable to find type of resource for conversion.
Explanation
The _XmTypedArgToArg function detected that the passed resource type in the typed argument list did not match a known resource type.
System action
The application continues.
Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
VaSimple.c
EZYXS63  Type conversion failed.
Explanation
The \_XmTypedArgToArg function detected that the address of the receiving XrmValue variable was NULL. The XtConvert process failed.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VaSimple.c

EZYXS71 Name: string Class: string

Explanation
Motif Generic Header message called within VendorS.c for use with applications from the previous release of Motif.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
VendorS.c

EZYXS72 Action invoked with the wrong number of parameters.

Explanation
A function was called with one of the following:
- No arguments
- More than one argument
• An argument that was not valid

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
FileSync.c, Notebook.c, ScrollBar.c, SelectioB.c

**EZYXS81**
XmNtag cannot be NULL. Setting to empty string.

**Explanation**
The ValidateTag function detected that the XmNtag was set to NULL.

**System action**
XmNtag is set to default and the application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xm.a library.

**Procedure name**
XmRenderT.c

**EZYXS82**
Display is NULL. Cannot load font.

**Explanation**
The ValidateAndLoadFont function detected that there was no valid Display to which to load the font.

**System action**
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
XmRenderT.c

EZYXS83  XmNfontType invalid. Cannot load font.
W

Explanation
The ValidateAndLoadFont function detected that the specified font or fontset is not a valid type.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
XmRenderT.c

EZYXS84  Conversion failed. Cannot load font.
W

Explanation
The ValidateAndLoadFont function detected that the font conversion failed because no callback was issued. The font did not load.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xm.a library.

Procedure name
XmRenderT.c

**EZYXS85**  
XmNfontType set to XmAS_IS. Cannot load font.

**Explanation**
The ValidateAndLoadFont function detected that the fontType was set to XmAS_IS. The specified font is not loaded.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
XmRenderT.c

**EZYXS86**  
XmNloadModel is XmLOAD_IMMEDIATE but XmNfont and XmNfontName not specified. Cannot load font.

**Explanation**
The ValidateAndLoadFont function detected that the font and font name were not specified. The font was not loaded.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.
Procedure name
XmRenderT.c

EZYXS91 No font found.

Explanation
The OptLineMetrics function or the SpecifiedSegmentExtents function detected that a font was not specified.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
XmString.c

EZYXT01W Tab value cannot be negative.

Explanation
The XmTabCreate function or the XmTabSetValue function detected that the _XmTabValue contained a negative value.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
XmTabList.c

EZYXT11W XmNtextField resource cannot be set.
Explanation
The SetValues function detected that the current textfield Widget is not equal to the new textfield Widget, and cannot be set to the specified value.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
SSpinB.c

EZYXT12W XmNpositionType resource can only be set at creation time.

Explanation
The SetValues function detected that the current position of XmPOSITION_ARRAY,VALUE does not equal the new position. The position cannot be changed.

System action
The application continues.

Operator response
Contact the system programmer.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
SSpinB.c

EZYXT13W Item does not exist. XmNposition is unchanged.

Explanation
The XmSimpleSpinBoxSetItem function attempted to access an XmNposition value that is no longer present.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xm.a library.

Procedure name
SSpinB.c

EZYXU01 The duplicate option string was ignored
Explanation
The same command line option has been repeated more than once.

System action
The compiler continues.

Operator response
Remove duplicate command line option.

System programmer response
None.

Module
UilCmd.c

Procedure name
cmd_decode_command_line

EZYXU02 The unknown option string was ignored
Explanation
An unknown option has been used in the compiler command line.

System action
The compiler continues.

Operator response
Check what you typed on the command line.

System programmer response
None.
Module
UilCmd.c

Procedure name
cmd_decode_command_line

EZYXU03E  The additional UIL source file: string was ignored

Explanation
More than one source file was specified. Only the first source file will be compiled.

System action
The compiler continues.

Operator response
Compile additional source files using separate invocations or the compiler.

System programmer response
None.

Module
UilCmd.c

Procedure name
cmd_decode_command_line

EZYXU04S  An error occurred opening source file: string.

Explanation
The source file specified could not be opened.

System action
The compiler ends.

Operator response
Verify that the file exists and has the proper permissions. Invoke the compiler again.

System programmer response
None.

Module
UilDB.c, UilSrcSrc.c

Procedure name
various

EZYXU05S  An error occurred reading next line of source file: string.
**Explanation**  
The source file specified could not be successfully read.

**System action**  
The compiler ends.

**Operator response**  
Verify that the file exists and has the proper permissions. Invoke the compiler again.

**System programmer response**  
None.

**Module**  
UilDB.c, UilSrcSrc.c

**Procedure name**  
various

<table>
<thead>
<tr>
<th>EZYXU06S</th>
<th>An internal error occurred in: string</th>
</tr>
</thead>
</table>

**Explanation**  
The compiler detected an internal error.

**System action**  
The compiler ends.

**Operator response**  
Submit a software problem report.

**System programmer response**  
None.

**Module**  
UilDiag.c

**Procedure name**  
diag_handler

<table>
<thead>
<tr>
<th>EZYXU07E</th>
<th>The line was truncated at number characters</th>
</tr>
</thead>
</table>

**Explanation**  
The compiler encountered a source line greater than 132 characters. Characters beyond the 132 character limit were ignored.

**System action**  
The compiler continues.
**Operator response**
Break each source line longer than 132 characters into several source lines. Long string literals can be created using the concatenation operator.

**System programmer response**
None.

**Module**
UilSrcSrc.c

**Procedure name**
src_get_source_line

**EZYXU08E**  The value of string is out of range string.

**Explanation**
The value specified is outside the legal range for its type.

**System action**
The compiler continues.

**Operator response**
Change the UIL module source.

**System programmer response**
None.

**Module**
UilLexAna.c, UilSarMod.c, UilSemVal.c

**Procedure name**
various

**EZYXU09E**  string1 not terminated string2

**Explanation**
A string was not properly terminated. String1 and string2 explain the defect.

**System action**
The compiler continues.

**Operator response**
Change the UIL module source to terminate the sequence properly.

**System programmer response**
None.
EZYXU10E  The unprintable character char ignored

Explanation
The compiler encountered an illegal control character in the UIL specification file. The decimal value of the character is displayed.

System action
The compiler continues.

Operator response
Replace the character with the sequence specified in the message.

System programmer response
None.

EZYXU11E  The unknown sequence string ignored

Explanation
The compiler detected a sequence of printable characters it did not understand. The compiler omitted the sequence of characters shown.

System action
The compiler continues.

Operator response
Fix the UIL module source.

System programmer response
None.

EZYXU12E  The unknown escape sequence \char - \ ignored
Explanation
A back slash was followed by an unknown escape character. The back slash is the escape character in UIL. A selected set of single characters can follow a back slash such as \n for new line. The character following the back slash was not one of the selected set.

System action
The compiler continues.

Operator response
Fix the UIL module source.

System programmer response
None.

Module
UilLexAna.c

Procedure name
yylex

EZYXU13E  The name exceeds number characters. It was truncated to: string

Explanation
The UIL compiler encountered a name longer than 31 characters. The compiler truncated the name to the leftmost 31 characters.

System action
The compiler continues.

Operator response
Fix the UIL module source.

System programmer response
None.

Module
UilLexAna.c

Procedure name
yylex

EZYXU14S  The compiler ran out of virtual memory

Explanation
The compiler ran out of virtual memory.

System action
The compiler continues.
**Operator response**
Reduce the size of your application or increase virtual memory.

**System programmer response**
None.

**Module**
UilMain.c

**Procedure name**
UIL_mmove

**EZYXU15E** An unexpected string token was seen. Parsing will resume after string.

**Explanation**
At the point marked in the module, the compiler found a construct, such as a punctuation mark, name, or keyword, when it was expecting a different construct. The compiler continued analyzing the module at the next occurrence of the construct stated in the message.

**System action**
The compiler continues.

**Operator response**
Check the syntax of your UIL module at the point marked by the compiler. If the module specifies case-sensitive names, check that your keywords are in lowercase characters.

**System programmer response**
None.

**Module**
UilLexAna.c, UilLexPars.c

**Procedure name**
various

**EZYXU16E** string string must be defined before this reference

**Explanation**
The widget pointed to in the message was either never defined or not defined prior to this point in the module. The compiler requires to be defined before you see the widget at the next occurrence of the construct stated in the message.

**System action**
The compiler continues.

**Operator response**
Check for a misspelling of the name of the widget, a missing declaration for the widget, or declaring the widget after its first reference. If names in the module are case-sensitive, the spellings of the name in the declaration and in the reference must match exactly.
**EZYYU17E** The context requires a string but *string* was specified

**Explanation**
At the point marked in the specification, one type of object (such as a widget) is required and your specification supplied a different type of object (such as a value).

**System action**
The compiler continues.

**Operator response**
Check for misspelling. Also check to ensure that you have referred to the intended widget.

**EZYYU18E** *string* is not implemented yet

**Explanation**
You are using a feature of UIL that has not been implemented yet.

**System action**
The compiler continues.

**Operator response**
Try an alternate technique.

**System programmer response**
None.
**EZYXU19E**  *string value was found when string value was expected*

**Explanation**
The indicated value is not of the specific type required by UIL in this context.

**System action**
The compiler continues.

**Operator response**
Check the definition of the function or clause.

**System programmer response**
None.

**Module**
UilSarVal.c

---

**EZYXU20W**  The *string string* is not supported for the *string object*

**Explanation**
Each widget or gadget supports a specific set of parameters, reasons, and children. The particular parameter, reason, or child you specified is not supported for this widget or gadget.

**System action**
The compiler continues.

**Operator response**
If a widget creation function accepts a parameter that UIL rejects, it does not necessarily mean that the UIL compiler is in error. Widget creation functions ignore parameters that they do not support without notifying you that the parameter is being ignored.

**System programmer response**
None.

**Module**
UilSemVal.c

---

**EZYXU21I**  This *string string* supersedes a previous definition in this *string string*
Explanation
A parameter or call-back list has either a duplicate parameter or duplicate reason.

System action
The compiler continues.

Operator response
This is not necessarily an error. The compiler is alerting you to make sure that you intend to override the value of a prior parameter. This informational message can be suppressed using the -I option.

System programmer response
None.

Module
UilSemVal.c, UilSarMod.c

Procedure name
various

EZYXU22E  The name string previously defined as string

Explanation
The name marked by the message was used in a previous declaration. UIL requires that the names of all widgets declared within a module be unique.

System action
The compiler continues.

Operator response
Check for a misspelling. If the module is case-sensitive, the spellings of names in declarations and in references must match exactly.

System programmer response
None.

Module
UilSarVal.c

Procedure name
sem_dcl_name

EZYXU23E  The value used in this context must be private

Explanation
A private value is one that is not imported or exported. In the context marked by the message, only a private value is legal. Situations where this message is issued include defining one value in terms of another, and defining parameters in terms of functions. In general, a value must be private when the compiler must know the value at compilation time. Exported values are disallowed in these context, even though a value is present, because that value could be overridden at run time.
**System action**
The compiler continues.

**Operator response**
Change the value to private.

**System programmer response**
None.

**Module**
UilSarVal.c, UilSemVal.c

**Procedure name**
various

**EZYXU24E**  The procedure *string* was previously declared with *number* arguments

**Explanation**
The declaration of the marked procedure specified a different number of parameters than are present in this procedure reference.

**System action**
The compiler continues.

**Operator response**
Check that you are calling the correct function. If you intend to call the procedure with a varying number of parameters, omit the argument list in the procedure declaration.

**System programmer response**
None.

**Module**
UilSemVal.c

**Procedure name**
sem_validate_procref_entry

**EZYXU25E**  *string* value was found. The argument to procedure *string* must be *string* value

**Explanation**
The declaration of the marked procedure specified a different type of parameters than is present in this procedure reference.

**System action**
The compiler continues.
Operator response
Check that you are passing the correct parameter to the correct function. If you intend to call the procedure with varying parameter types, declare the procedure specifying 'any' as the type of the parameter.

System programmer response
None.

Module
various

Procedure name
various

EZYXU26E  string string was found when string string was expected

Explanation
Most parameters take values of a specific type. The value specified is not correct for this procedure.

System action
The compiler continues.

Operator response
The message indicates the expected type of parameter. Check that you have specified the intended value and that you specified the correct parameter.

System programmer response
None.

Module
UilSarMod.c, UilSarObj.c

Procedure name
various

EZYXU27E  string string was never defined

Explanation
Certain UIL objects, such as gadgets and widgets, can be referred to before they are defined. The marked widget is such an object, however the compiler never found the widget declaration.

System action
The compiler continues.

Operator response
Check for misspelling. If the module is case-sensitive, the spellings of names in declarations and in references must match exactly.
System programmer response
None.

Module
UilP2Reslv.c

Procedure name
sem_resolve_forward_ref

EZYXU28E string string was already specified

Explanation
A widget or gadget declaration can have, at most, one argument list, one call-back list, and one controls list.

System action
The compiler continues.

Operator response
If you want to specify multiple lists of arguments, controls and call-backs, you can do so within one list. For example: arguments (argument_list1; argument_list2;).

System programmer response
None.

Module
UilSarObj.c

Procedure name
sar_save_feature

EZYXU29E string item is not allowed in string string

Explanation
The indicated list item is not of the type required by the list. Argument lists must contain argument entries, call-back lists must contain call-back entries, control lists must contain control entries and procedure lists must contain procedure entries.

System action
The compiler continues.

Operator response
Check the syntax for the type of list entry that is required in this context and change the indicated list item.

System programmer response
None.

Module
UilSarObj.c, UilSemVal.c
Procedure name
various

EZYXU30S  The compilation was terminated. Fix the previous errors

Explanation
Errors encountered during the compilation have caused the compiler to stop processing.

System action
The compiler ends.

Operator response
Fix the errors already diagnosed by the compiler and re-compile.

System programmer response
None.

Module
UilDiags.c

Procedure name
diag_issue_diagnostic

EZYXU31S  An internal error occurred. Submit a defect report.

Explanation
The compiler diagnosed an internal error.

System action
The compiler ends.

Operator response
Get a listing and identify where the error is being issued. Try to correct any faulty syntax in this area. If you are unable to prevent this error, submit a software problem report.

System programmer response
None.

Module
UilDiags.c

Procedure name
diag_issue_diagnostic

EZYXU33E  string was missing following "string" option

Explanation
You used a command line option that requires a parameter and you did not provide that parameter.
System action
The compiler continues.

Operator response
Omit the option or provide the parameter.

System programmer response
None.

Module
UilCmd.c

Procedure name
cmd_decode_command_line

EZYXU34S  An error occurred opening the listing file: filename

Explanation
The compiler could not create the listing file noted in the message.

System action
The compiler ends.

Operator response
Check that you have write access to the directory you specified to hold the listing file.

System programmer response
None.

Module
UilLstLst.c

Procedure name
lst_open_listing

EZYXU35S  An error occurred writing to the listing file: filename

Explanation
The compiler could not write a line into the listing file noted in the message.

System action
The compiler ends.

Operator response
Check to see that there is adequate space in the disk specified to hold the listing file.
System programmer response
None.

Module
UilLstLst.c

Procedure name
lst_output_line

EZYXU36E  An invalid module structure was detected. Check the UIL module syntax

Explanation
The structure of the UIL module is incorrect.

System action
The compiler continues.

Operator response
If there are any syntax errors reported, correct them and re-compile.

System programmer response
None.

Module
UilMain.c

Procedure name
common_main

EZYXU37S  Too many source files are open: number

Explanation
The compiler has a fixed limit for the number of source and include files that it can process. This number is reported in the message.

System action
The compiler ends.

Operator response
Use fewer include files.

System programmer response
None.

Module
UilSrcSrc.c
**Procedure name**
src_open_file

**EZYXU39I**  errors: number warnings: number informationals: number

**Explanation**
This message lists a summary of the diagnostic data issued by the compiler and is displayed only when diagnostic data has been issued.

**System action**
The compiler continues.

**Operator response**
Correct the problems reported. You can use the -I option to suppress informational and warning diagnostic data that you have determined to be harmless.

**System programmer response**
None.

**Module**
UilDiags.c

**Procedure name**
diag_issue_summary

**EZYXU40S**  An error occurred opening the UID file: filename

**Explanation**
The compiler could not create the UID file noted in the message. A UID file holds the compiled user-interface specification.

**System action**
The compiler ends.

**Operator response**
Check that you have write access to the directory you specified to hold the UID file.

**System programmer response**
None.

**Module**
UilP2Out.c

**Procedure name**
sem_output_uid_file

**EZYXU41I**  No UID file was produced
Explanation
If the compiler reported recoverable or nonrecoverable errors, no UID file is produced. The message informs you that the compiler did not produce a UID file.

System action
The compiler continues.

Operator response
Fix the problems reported by the compiler.

System programmer response
None.

Module
UilP2Out.c

Procedure name
sem_output_uid_file

EZYXU42E  The creation procedure is not supported by the string widget

Explanation
You specified a creation procedure for a toolkit widget. You can specify a creation procedure only for a user-defined widget.

System action
The compiler continues.

Operator response
Remove the procedure clause following the widget type.

System programmer response
None.

Module
various

Procedure name
various

EZYXU43E  The creation procedure is not allowed in a string widget reference

Explanation
You specified a creation procedure when referencing a widget. You can specify a creation procedure only when you declare the widget.

System action
The compiler continues.
Operator response
Remove the procedure clause following the object type.

System programmer response
None.

Module
UilSarObj.c

Procedure name
sar_verify_object

EZYXU44E The creation procedure is required in a string widget declaration

Explanation
When defining a user-defined widget, you must specify the name of the creation function for creating an instance of this widget.

System action
The compiler continues.

Operator response
Insert a procedure clause following the widget type in the widget declaration. You also need to declare the creation procedure using a procedure declaration.

System programmer response
None.

Module
UilSarObj.c

Procedure name
sar_verify_object

EZYXU45 A NULL character in a string is not supported

Explanation
You have created a string that has an embedded null character. Strings are represented in a UID file and in many toolkit data structures as null-terminated strings. So, although the embedded nulls will be placed in the UID file, toolkit functions might interpret an imbedded null as the terminator for the string.

System action
The compiler continues.

Operator response
Be careful when using embedded nulls.
System programmer response
None.

Module
UilLexAna.c

Procedure name
yyparse

EZYXU46E  Widget string is part of a circular definition

Explanation
The indicated object is referenced as a descendant of itself, either within its own definition or within the definition of one of the widgets in the widget tree that the object controls.

System action
The compiler continues.

Operator response
Change the definition of the indicated widget so that it is not a descendant of itself.

System programmer response
None.

Module
UilSemVal.c

Procedure name
sem_validate_callback_entry

EZYXU47S  No source file was specified

Explanation
No source file was specified to compile.

System action
The compiler ends.

Operator response
Specify the name of a UIL specification file to compile.

System programmer response
None.
Procedure name
cmd_decode_command_line

EZYXU48  string string supports only a single string string

Explanation
You have specified a particular clause more than once in a context where that clause can only occur once. For example, the version clause in the module can only occur once.

System action
The compiler continues.

Operator response
Choose the correct clause and delete the others.

System programmer response
None.

Module
UilSarMod.c

Procedure name
various

EZYXU49  string widget supports only a single control

Explanation
The specified widget might only have one entry in its controls list.

System action
The compiler continues.

Operator response
Change the control list to have only one entry.

System programmer response
None.

Module
UilP2Out.c

Procedure name
out_emit_widget

EZYXU51E  Place the names clause before other module clauses
Explanation
The case-sensitive clause, if specified, must be the first clause following the name of the module. You have inserted another module clause before this clause.

System action
The compiler continues.

Operator response
Reorder the module clauses so that the case-sensitivity clause is first.

System programmer response
None.

Module
UilSarMod.c

Procedure name
sar_process_module_sensitivity

EZYXU52E The color letter string must be a single character

Explanation
The string associated with each color in a color table must hold exactly one character. You have specified a string with either fewer or more characters.

System action
The compiler continues.

Operator response
Use a single character to represent each color in a color table.

System programmer response
None.

Module
UilSarVal.c

Procedure name
sar_make_color_item

EZYXU53E The color letter was used for prior color in this table

Explanation
Each of the letters used to represent a color in a color table must be unique. If not, that letter in an icon would represent more than one color. The letter marked has been assigned to more than one color.

System action
The compiler continues.
Operator response
Use a single character to represent each color in a color table.

System programmer response
None.

Module
UilSarVal.c

Procedure name
sar_make_color_item

EZYXU54E Row number must have same width as row 1

Explanation
The icons supported by UIL are rectangular. As a result, each of the strings used to represent a row of pixels in an icon must have the same length. The specified row does not have the same length as the first row.

System action
The compiler continues.

Operator response
Use a single character to represent each color in a color table.

System programmer response
None.

Module
UilSarVal.c

Procedure name
sar_make_icon

EZYXU55E row number, column number: letter letter is not in the color table

Explanation
You have specified a color to be used in an icon that is not in the color table for that icon. The invalid color is identified in the message by displaying the letter used to represent it.

System action
The compiler continues.

Operator response
Either add the color to the color table for that icon, or use a character representing a color in the color table.

System programmer response
None.
Module
UilSemVal.c

Procedure name
sem_evaluate_value_expr

EZYXU56E  There are too many string in string. The limit is number

Explanation
You exceeded a compiler limit such as the number of fonts in a font table or the number of strings in translation table. The message indicates the limit imposed by the compiler.

System action
The compiler continues.

Operator response
Restructure your UIL module.

System programmer response
None.

Module
UilCmd.c, UilSarVal.c

Procedure name
various

EZYXU58  The string gadget is not supported. The string widget will be used instead

Explanation
The indicated widget type does not support a gadget variant, only a widget variant is supported for this widget type. The UIL compiler ignores the gadget indication and creates widgets of this widget type.

System action
The compiler continues.

Operator response
Restructure your UIL module.

System programmer response
None.

Module
UilSarMod.c, UilSarObj.c

Procedure name
various
EZYXU59E  The string type is not valid for string

Explanation
The indicated operand is not of a type that is supported by this operator.

System action
The compiler continues.

Operator response
Check the definition of the operator and make sure the type of the operand you specify is supported by the operator.

System programmer response
None.

Module
UilSemVal.c

Procedure name
validate_arg

EZYXU61  The string constraint is not supported for the string string

Explanation
You have specified a constraint that does not exist or is not appropriate for the constrained object.

System action
The compiler continues.

Operator response
Check for spelling errors. Check that the constraint is appropriate.

System programmer response
None.

Module
UilSemVal.c

Procedure name
sem_validate_constraint_entry

EZYXU62  Too many string options were detected, the limit is number

Explanation
You have specified more directories (using command line options) than the UIL compiler can process.
System action
The compiler continues.

Operator response
Reduce the number of directories specified by command line options.

System programmer response
None.

Module
UilCmd.c

Procedure name
cmd_decode_command_line

EZYXU63 An error occurred while closing the source file: filename

Explanation
The UIL compiler was not able to properly close a source file.

System action
The compiler continues.

Operator response
Check the permissions of the file and its parent directory.

System programmer response
None.

Module
UilSrcSrc.c

Procedure name
Uil_src_cleanup_source

EZYXU64E The string value is circularly defined

Explanation
You attempted to declare a value using an expression that contains the value you are declaring.

System action
The compiler continues.

Operator response
Remove the circular reference from the value expression.
System programmer response
None.

Module
UilSemVal.c

Procedure name
various

EZYXU65 The string built-in name was overridden
W

Explanation
You have declared the meaning of a nonreserved keyword again. This is permitted, but generates a warning. You will not be able to use the UIL-supplied function for the keyword after it is overridden.

System action
The compiler continues.

Operator response
Make sure that you intended to override the built-in keyword. Also make sure that you no longer need the UIL-supplied function for that keyword.

System programmer response
None.

Module
UilSarVal.c

Procedure name
sem_dcl_name

EZYXU66 The string argument does not support enumerated values
W

Explanation
You used the displayed argument along with a type that the argument does not support.

System action
The compiler continues.

Operator response
Correct the argument function. Use only supported types for the argument.

System programmer response
None.
Module
UilSemVal.c

Procedure name
sem_validate_argument_enumset

**EZYXU67** The string argument does not support the string enumerated value

**Explanation**
You used the displayed argument along with an inappropriate value. Review the allowed values for the resources supported by the widget.

**System action**
The compiler continues.

**Operator response**
Correct the argument function. Use only supported values for the argument.

**System programmer response**
None.

Module
UilSemVal.c

Procedure name
sem_validate_argument_enumset

**EZYXU68E** The environment variable, $LANG, contains an unknown character set

**Explanation**
The $LANG environment variable contains an unknown character set. The UIL compiler does not have a default definition for the character set, and you have not supplied a definition using the CHARACTER_SET function.

**System action**
The compiler continues.

**Operator response**
Make sure the character set is spelled correctly. If it is a user-defined character set, make sure you have defined it within a CHARACTER_SET function.

**System programmer response**
None.
Procedure name
lex_initialize_analyzer

EZYXU69E The string object’s controls hierarchy contains a reference to itself

Explanation
You have referred to a widget within its own controls list. Widgets cannot act as their own controlled descendants.

System action
The compiler continues.

Operator response
Remove the circular reference from the widget controls list.

System programmer response
None.

Module
UilSemVal.c

Procedure name
various

EZYXU70S The value string is too large for context buffer

Explanation
The UIL compiler was unable to allocate an internal buffer while attempting to parse the specified value.

System action
The compiler ends

Operator response
If possible, reduce the complexity of the value expression.

System programmer response
None.

Module
UilP2Out.c

Procedure name
out_emit_value

EZYXU71S Forward referencing is not allowed for string

Explanation
You have referenced an object before defining it.
System action
The compiler ends

Operator response
Restructure the module so the object is defined before being referenced.

System programmer response
None.

Module
UilSarMod.c

Procedure name
sar_process_module_version

**EZYXU72E**  *string* type cannot be converted to *string* type

Explanation
You have defined an expression containing two types of operands that cannot be converted into one another (for example, integer and Pixmap) or that require an explicit conversion function (for example, Boolean and Floating Point).

System action
The compiler continues

Operator response
Make sure that the expression is correct and that the required conversion functions are used.

System programmer response
None.

Module
UilSemVal.c

Procedure name
various

**EZYXU73E**  *string* is invalid

Explanation
You defined an argument function using an argument that UIL does not allow.

System action
The compiler continues

Operator response
Make sure the argument function is correct.
System programmer response
None.

Module
UilSemVal.c

Procedure name
various

**EZYXU74S**  An error occurred while reading the binary database

Explanation
The binary database file could not be successfully read.

System action
The compiler ends

Operator response
Verify that the file exists and has the proper permissions. Invoke the compiler again.

System programmer response
None.

Module
UilDB.c

Procedure name
various

**EZYXU75S**  The binary database was compiled with a future version

Explanation
The binary database was compiled by a UIL version later than the compiler version being used.

System action
The compiler ends

Operator response
Use a binary database compiled with the current UIL compiler. Invoke the compiler again.

System programmer response
None.
Procedure name

db_incorporate

EZYXU76S   An error occurred while opening the database file: filename

Explanation
The binary database file could not be opened.

System action
The compiler ends

Operator response
Verify that the file exists and has the proper permissions. Invoke the compiler again.

System programmer response
None.

Module
UilDB.c

Procedure name
db_open_file

EZYXU77S   An error occurred while writing to the UID file: filename

Explanation
The compiler could not write a line into the UID file noted in the message.

System action
The compiler ends

Operator response
Check to see that there is adequate space in the disk specified to hold the UID file.

System programmer response
None.

Module
UilP2Out.c

Procedure name
various

EZYXU79S   Invalid include file name

Explanation
The value_entry structure is defined as a type other than char_8, when char_8 is required.
**System action**
The compiler ends

**Operator response**
Ensure usage of the correct structure type.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
UilSarInc.c

**Procedure name**
sar_include_file()

**EZYXU80**  Incompatible unit types for arithmetic operation

**Explanation**
While performing an arithmetic operation with multiple arguments, the compiler found that the arguments were of incompatible types. For example, the compiler was expecting to perform an AND operation on two integer variables, and found that one variable was a floating point.

**System action**
The compiler ends

**Operator response**
Verify that the UID file has Integer and FloatingPoint Values in the correct order or place.

**System programmer response**
None.

**Module**
UilSemVal.c

**Procedure name**
sem_evaluate_value_expr()

**EZYXU81I**  string used as charset name; string used as charset component

**Explanation**
During NAME to CHARSET_NAME conversion, the specified charset could not be resolved. The default charset was used.

**System action**
None.
Operator response
If the default charset is not acceptable, verify that the correct charset is accessible and correctly labeled.

System programmer response
None.

Module
UilSarComp.c

Procedure name
sar_make_fallback_charset()

EZYXU82E  string string already specified for this string string

Explanation
A widget or gadget declaration can have no more than one argument list, one call-back list, and one control list.

System action
The compiler continues.

Operator response
If you want to specify multiple lists of arguments, controls and call-backs, you can do so within one list. For example: arguments (argument_list1; argument_list2;).

System programmer response
None.

Module
UilSarObj.c

Procedure name
sar_save_feature()

EZYXU99E  string1 string2 string3

Explanation
The Mrm function, string1, has detected an error. string2 describes the error. string3 states the status returned by the Mrm function.

System action
None.

Operator response
None.
System programmer response

Based on the error description, review the application code and the UIL module. Verify that the correct UID module is being accessed. In most cases, this is an application programming error. If the error description states that this is an internal error, submit a software problem report.

Module
Mrerror.c

Procedure name
Urm__UT_Error

---

**EZYXW01**  
Xlib: connection to string refused to server

---

**Explanation**
The server identified by string has refused the connection.

**System action**
The application is ended.

**Operator response**
Issue the xhost command at the server to add the host address.

---

System programmer response
None.

---

**Module**
OpenDis.c

**Procedure name**
XOpenDisplay

---

**EZYXW02**  
Xlib: client uses different protocol version (number) than server (number)!

---

**Explanation**
There is a mismatch between the version of the X Window System protocol used by the client library and that used by the server (string2)

**System action**
The application is ended.

**Operator response**
Use a server which is compatible with the client library.

---

System programmer response
None.
Module
OpenDis.c

Procedure name
XOpenDisplay

**EZVXW03**
Error parsing argument "string1" (string2); string3

Explanation
The Resource Manager has been passed an invalid argument. String1 is the option name; string2 is the value supplied; string3 is an explanatory comment.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
ParseCmd.c

Procedure name
_XReportParseError

**EZVXW04**
X connection to string broken (explicit kill or server shutdown).

Explanation
The server has been shut down or the client window has been destroyed. This message might be generated when a window is closed by the user and the application does not recognize what has happened.

System action
The application is ended.

Operator response
None.

System programmer response
None.
Explanation
The connection to the server has been broken. The reported error is number1 with a descriptive string string1. String2 identifies the server being used.

System action
The application is ended.

Operator response
Try to restart the application.

System programmer response
None.

Module
XlibInt.c

Procedure name
_XDefaultIOError

Explanation
An error has been detected by an X Window System extension. String1 is the name of the extension; string2 is a description of the error; String3 is the address of the server.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
extutil.c

Procedure name
default_exterror

Warning: Current locale is not supported by Xlib
Explanation
The current locale is not supported by Xlib. The server code set is set to ISO8859-1.

System action
The application continues.

Operator response
None.

System programmer response
None.

Module
Conv.c

Procedure name
init_iconv

EZYXW08 Warning: Unable to convert from string1 to string2.

Explanation
Conversion tables do not exist for converting from code set string1 to code set string2. The default conversions are used.

System action
The application continues.

Operator response
None.

System programmer response
None.

Module
Conv.c

Procedure name
init_iconv

EZYXW09I Unable to open message catalog: X11R6.cat

Explanation
The message catalog for X Window System messages could not be opened.

System action
The application continues.
Operator response
Verify that the NLSPATH environment variable is set to the correct value.

System programmer response
None.

Module
OpenDis.c, Xt/Initialize.c

Procedure name
XOpenDisplay,XtToolitInitialize

EZYXW10  E  X Error of failed request: string Major op code of failed request: number string (additional lines depending on X Error) Serial number of failed request: number Current serial number in output stream: number

Explanation
This message is displayed by the default Xlib error handler. An X Window System protocol error has been detected. The type of error detected is described by string. The failed request Major op code is displayed as a number and as a descriptive string. This is followed by one or more of the following lines, depending on the type of error: ResourceID in failed request: hexadecimal number AtomID in failed request: hexadecimal number value in failed request: hexadecimal number Minor code of failed request: number. These lines are followed by the serial number of the failed request and the current serial number.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
XlibInt.c

Procedure name
_XPrintDefaultError

EZYXW11  E  XtlibError: string1 string2 string3 string4

Explanation
An Xtlib function has detected an error. String1 is the name of the function reporting the error, string2 is the error type, string3 is the error class, and string4 is a descriptive string.

System action
The application is ended.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
XtMsgCat.c

Procedure name
XtErrorMsgCat

EZYXW12  X11libWarning: string1 string2 string3 string4
W

Explanation
An X11lib function has detected a recoverable error. String1 is the name of the function reporting the error, string2 is the error type, string3 is the error class, and string4 is a descriptive string.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
XtMsgCat.c

Procedure name
XtWarningMsgCat

EZYXW13  X11lib: locale not supported by C library, locale unchanged.
W

Explanation
An attempt was made to set the current locale to a value not supported by the C library.

System action
The application continues.

Operator response
None.

System programmer response
None.
**Module**
Initialize.c

**Procedure name**
_XtDefaultLanguageProc

**Explanation**
An attempt was made to set the current locale to a value not supported by Xlib.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.

---

**Module**
Initialize.c

**Procedure name**
_XtDefaultLanguageProc

**Explanation**
The locale modifiers could not be set.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
None.
EZYXW16  XtVaTypedArg is not valid in XtVaSetSubvalues()

Explanation
XtVaSetSubvalues has been called with typed arguments.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
VarCreate.c

Procedure name
XtVaSetSubvalues

EZYXW17  XtVaTypedArg is an invalid argument to XtVaGetSubvalues()

Explanation
XtGetSubvalues has been called with typed arguments.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
VarGet.c

Procedure name
XtVaGetSubvalues

EZYXW18  String to BackingStore conversion needs no extra arguments

Explanation
The string to backing store conversion has been called with an extra argument.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xmu/StrToBS.c

Procedure name
XmuCvtStringToBackingStore

**EZYXW19** Xawlib: Too many parameters passed to highlight action table.

Explanation
Extra parameters were passed in a call to Highlight().

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/Command.c

Procedure name
Highlight

**EZYXW20** List Widget: Unknown geometry return.

Explanation
A call to ChangeSize contained an unknown geometry request.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/List.c

Procedure name
ChangeSize

EZYXW21 List Widget: Size changed when it shouldn't have when resizing.

Explanation
An attempt was made to resize the List widget inappropriately.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/List.c

Procedure name
Resize

EZYXW22 MenuButton: Could not find menu widget named string.

Explanation
An incorrect name was passed for a menu widget.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
MenuButton.c
Procedure name: PopupMenu

**EZYXW23**  MultiSrc: The XtNuseStringInPlace resources may not be changed.

**Explanation**
An attempt was made to change the XtNuseStringInPlace resource.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xaw/MultiSrc.c

---

Procedure name: SetValues

**EZYXW24**  string1: unable to allocate number bytes for widget name

**Explanation**
Unable to allocate the required number of bytes for a widget name. String1 is the name of the routine allocating the storage.

**System action**
The application is ended.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xmu/WidgetNode.c

---

Procedure name: XmuWnInitializeNodes

**EZYXW25**  String1: unable to calloc number1 number2 byte widget node ptrs

---

EZYXxxx messages 1033
Explanation
Unable to allocate number1 number2 byte areas of storage for use as widget node pointers.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xmu/WidgetNode.c

Procedure name
XmuWnFetchResources

EZYXW26 XawTextWidget: An attempt was made to insert an illegal selection.

Explanation
An attempt was made to insert an illegal selection into an Xaw Text widget.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
TextAction.c

Procedure name
StartAction

EZYXW27 Xaw MultiSrc Object: possible memory leak in FreeAllPieces().

Explanation
A possible memory leak has been detected in FreeAllPieces().

System action
The application continues.
**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
MultiSrc.c

**Procedure name**
FreeAllPieces

**EZYXW28** Paned GripAction(); 1st parameter invalid.

**Explanation**
The first parameter passed to HandleGrip is invalid.

**System action**
The application is ended.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xaw/Paned.c

**Procedure name**
HandleGrip

**EZYXW29** Scrollbar Widget: Could not get geometry of thumb pixmap.

**Explanation**
The scrollbar widget could not get the geometry of the thumb pixmap.

**System action**
The application is ended.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.
**Module**
Xaw/Scrollbar.c

**Procedure name**
CreateGC

---

**EZYXW30**

**string Widget: The Simple Widget class method 'change_sensitive' must be defined or inherited.**

**Explanation**
The widget named string does not have the Simple Widget class method 'change_sensitive' defined or inherited.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

---

**Module**
Xaw/Simple.c

**Procedure name**
ClassPartInitialize

---

**EZYXW31**

**Paned GripAction has been passed incorrect parameters.**

**Explanation**
HandleGrip has been passed incorrect parameters.

**System action**
The application is ended.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

---

**Module**
Xaw/Paned.c

**Procedure name**
HandleGrip
**EZYXW32**  Unknown event type in GetEventEntry().

**Explanation**
An unknown event type has been detected.

**System action**
The application is ended.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
SimpleMenu.c

**Procedure name**
GetEventEntry

---

**EZYXW33**  No Dynamic class change of the SimpleMenu Label.

**Explanation**
An attempt was made to change the class of the SimpleMenu label.

**System action**
The application continues.

**Operator response**
None.

**System programmer response**
Use your X Window System/Motif documentation to correct this application programming error.

**Module**
Xaw/SimpleMenu.c

**Procedure name**
GetValues

---

**EZYXW34**  Xaw - SimpleMenuWidget: position menu action expects only one parameter which is the name of the menu.

**Explanation**
There should be only one parameter passed to PositionMenuAction.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/SimpleMenu.c

Procedure name
PositionMenuAction

EZYXW35  Xaw - SimpleMenuWidget: could not find menu named: string

Explanation
The menu widget named string could not be found.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/SimpleMenu.c

Procedure name
PositionMenuAction

EZYXW36  Xaw Simple Menu Widget: label string is NULL or label already exists, no label is being created.

Explanation
The label string is NULL or the label already exists for a SimpleMenu widget.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/SimpleMenu.c

Procedure name
CreateLabel

EZYXW37  Xaw Simple Menu Widget: Could not find location of mouse pointer.

Explanation
The position of the mouse pointer could not be determined.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/SimpleMenu.c

Procedure name
PositionMenu

EZYXW38  Xaw SmeBSB Object: Could not get Left Bitmap geometry information for menu entry "string"

Explanation
The Left Bitmap geometry information for the menu entry named string could not be obtained.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/SmeBSB.c
Procedure name
GetBitmapInfo

EZYXW39 Xaw SmeBSB Object: Left Bitmap of entry "string" is not one bit deep.

Explanation
The bitmap specified for entry string is not one bit deep.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/SmeBSB.c

Procedure name
GetBitmapInfo

EZYXW40 Xaw SmeBSB Object: Could not get Right Bitmap geometry information for menu entry "string".

Explanation
The Right Bitmap geometry information for the menu entry named string could not be obtained.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/SmeBSB.c

Procedure name
GetBitmapInfo

EZYXW41 Xaw SmeBSB Object: Right Bitmap of entry "string" is not one bit deep.
Explanation
The bitmap specified for entry string is not one bit deep.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/SmeBSB.c

Procedure name
GetBitmapInfo

EZYXW42 W Xaw Text Widget string: Vertical scrolling not allowed with height resize. Vertical scrolling has been DEACTIVATED.

Explanation
An attempt was made to specify vertical scrolling with height resizing for Text widget string.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/Text.c

Procedure name
Initialize

EZYXW43 W Xaw Text Widget string: Horizontal scrolling not allowed with wrapping active. Horizontal scrolling has been DEACTIVATED.

Explanation
An attempt was made to specify horizontal scrolling when wrapping was active for Text widget string.

System action
The application continues.
Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/Text.c

Procedure name
Initialize

EZYXW44 Xaw Text Widget string: Horizontal scrolling not allowed with width resize. Horizontal scrolling has been DEACTIVATED.

Explanation
An attempt was made to specify horizontal scrolling and width resize for Text widget string.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/Text.c

Procedure name
Initialize

EZYXW45 Xaw Text Widget: empty selection array.

Explanation
The selection array passed to DoSelection is empty.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.
Module
Xaw/Text.c

Procedure name
DoSelection

EZYXW46 Xaw Text Widget: multiply() takes exactly one argument.
E

Explanation
An incorrect number or arguments was passed to multiply().

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/TextAction.c

Procedure name
Multiply

EZYXW47 Xaw Text Widget: multiply() argument must be a number greater than zero, or 'Reset'.
E

Explanation
The argument to multiply must be a number greater than zero, or Reset.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/TextAction.c

Procedure name
Multiply
string This action must have only one or two parameters.

Explanation
The action, string, must have only one or two parameters.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/TextPop.c

Procedure name
_XawTextSearch

string The first parameter must be either backward or forward.

Explanation
The first parameter for the Search routine of the Text Widget is not either 'backward' or 'forward'.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/TextPop.c

Procedure name
Search

string Toggle Widget Error - Attempting to create a new toggle group when one already exists.

Explanation
An attempt was made to create a new toggle group when one already exists.
System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/Toggle.c

Procedure name
CreateRadioGroup

EZYXW51 Aborting, due to errors resolving bindings in the Toggle widget.

Explanation
There is an error in the Toggle widget's action table.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/Toggle.c

Procedure name
ClassInit

EZYXW52 we can not open any input method

Explanation
Xaw is unable to open any input method.

System action
The application continues.

Operator response
None.
System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/XawIm.c

Procedure name
OpenIM

EZYXW53  Xaw: input method doesn't support any style
W

Explanation
The current input method does not support any style.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/XawIm.c

Procedure name
OpenIM

EZYXW54  Xaw: input method doesn't support my input style.
W

Explanation
The input method being opened does not specify any supported style.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xaw/XawIm.c
Procedure name
OpenIM

**EZYXW55**

**Error**

**E**

**XSMP error:** Offending minor code = number (string) Offending sequence number = number
Error class = string Severity = string (may be followed by: ) BadValue Offset = number
BadValue Length = number BadValue = number

**Explanation**

The session manager client routines have detected an error. The Offending minor code is displayed as a number and an explanatory string. The Error class and severity are displayed as strings. If the Error class is BadValue, the offset, length, and the bad value are displayed.

**System action**

The application is ended.

**Operator response**

None.

**System programmer response**

Use your X Window System/Motif documentation to correct this application programming error.

**Module**

SM/sm_error.c

Procedure name

_SmcDefaultErrorHandler

---

**EZYXW56**

**ICE error:** Offending minor code = number (string) Offending sequence number = number
Error class = string Severity = string (may be followed by one or more of the following:)
BadValue Offset = string BadValue Length = string BadValue = string Major opcode : number
Reason : string Protocol name : string

**Explanation**

ICE has detected an error. Information about the error is displayed.

**System action**

The application is ended.

**Operator response**

None.

**System programmer response**

Use your X Window System/Motif documentation to correct this application programming error.

**Module**

ICE/error.c

Procedure name

_IceDefaultErrorHandler
X Error: string

Explanation
An X Protocol error has been detected. String displays a description of the error.

System action
The application is ended.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
Xmu/DefErrMsg.c

Procedure name
XmuPrintDefaultErrorMessage

XSMP error: Offending minor code = number (string) Offending sequence number = number Error class = string Severity = string (may be followed by: ) BadValue Offset = number BadValue Length = number BadValue = number

Explanation
The session manager server routines have detected an error. The Offending minor code is displayed as a number and an explanatory string. The Error class and severity are displayed as strings. If the Error class is BadValue, the offset, length, and the bad value are displayed.

System action
The application continues.

Operator response
None.

System programmer response
Use your X Window System/Motif documentation to correct this application programming error.

Module
SM/sm_error.c

Procedure name
_SmcDefaultErrorHandler

Xaw Text widget string: empty copy queue.
**Explanation**  
An attempt was made to copy with an empty copy queue.

**System action**  
The application continues.

**Operator response**  
None.

**System programmer response**  
Use your X Window System/Motif documentation to correct this application programming error.

**Module**  
Xaw/Text.c

**Procedure name**  
PopCopyQueue

---

**EZYXW60I**  
Xlib: resource ID allocation space exhausted!

**Explanation**  
The X Window System resource ID space has been exhausted.

**System action**  
The application continues.

**Operator response**  
None.

**System programmer response**  
None.

**Module**  
XlibInt.c

**Procedure name**  
_XAllocID

---

**EZYXW61I**  
Xlib: sequence lost (number > number) in reply type number!

**Explanation**  
A sequence number has been received which is less than the last received sequence number.

**System action**  
The application continues.
Operator response
Data might have been lost, consider restarting the application.

System programmer response
None.

Module
XlibInt.c

Procedure name
_XSetLastRequestRead

EZYXW62I  Xlib: unexpected async reply (sequence number)!

Explanation
An asynchronous reply has been received when None. was expected.

System action
The application continues.

Operator response
Data might have been lost, consider restarting the application.

System programmer response
None.

Module
XlibInt.c

Procedure name
_XAsyncReply

EZYXW6eI  Xlib: unexpected connection setup reply from server, type 'nbr'.

Explanation
An unexpected reply has been received the server. This could be an unexpected authenticate reply.

System action
The application is ended.

Operator response
Verify that you are meeting the authentification requirements of the server you are connecting to. Try again.

System programmer response
None.
Module
OpenDis.c

Procedure name
XOpenDisplay

EZYXW63 Could not initialize the PEX extension on the specified display.
E

Explanation
The PEX server on the display could not be properly initialized.

System action
The application is ended.

Operator response
Verify that the PEX extension is active for the server.

System programmer response
None.

Module
PEX5/pl_startup.c

Procedure name
PEXInitialize

EZYXW64 Could not allocate memory for PEXlib internal usage.
E

Explanation
PEXlib could not allocate memory.

System action
The application is ended.

Operator response
Increase virtual memory size.

System programmer response
None.
Ezyxw65  Implicit call to PEXGetEnumTypeInfo by PEXInitialize failed.

Explanation
PEXlib could not initialize properly.

System action
The application is ended.

Operator response
Verify that the PEX extension is active for the server.

System programmer response
None.

Module
PEX5/pl_startup.c

Procedure name
PEXInitialize

Ezyxw66  No floating point formats supported by server.

Explanation
The PEX server extension does not support any floating point formats supported by the client.

System action
The application is ended.

Operator response
Use a server which supports the required floating point format.

System programmer response
None.

Module
PEX5/pl_startup.c

Procedure name
PEXInitialize

Ezyxw67  Could not get PEX extension information.

Explanation
Information about the PEX server extension could not be obtained.
System action
The application is ended.

Operator response
Verify that the PEX server extension is active.

System programmer response
None.

Module
PEX5/pl_startup.c

Procedure name
PEXInitialize

**EZYXW68** Client speaks PEX num.num; Server speaks PEX num.num

Explanation
Attempting to use incompatible PEX client and server.

System action
The application is ended.

Operator response
Verify that the correct PEX server extension is active.

System programmer response
None.

Module
PEX5/pl_startup.c

Procedure name
PEXInitialize

**EZYXW69I** Opcode of failed output command: number. Number of output commands processed: number.

Explanation
Displays the PEX opcode which failed.

System action
N/A

Operator response
None.
System programmer response
Correct the application and try again.

Module
PEX5/pl_startup.c

Procedure name
_PEXPrintOCError
Appendix A. Related protocol specifications

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:

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title and Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 652</td>
<td>Telnet output carriage-return disposition option D. Crocker</td>
</tr>
<tr>
<td>RFC 653</td>
<td>Telnet output horizontal tabstops option D. Crocker</td>
</tr>
<tr>
<td>RFC 654</td>
<td>Telnet output horizontal tab disposition option D. Crocker</td>
</tr>
<tr>
<td>RFC 655</td>
<td>Telnet output formfeed disposition option D. Crocker</td>
</tr>
<tr>
<td>RFC 657</td>
<td>Telnet output vertical tab disposition option D. Crocker</td>
</tr>
<tr>
<td>RFC 658</td>
<td>Telnet output linefeed disposition D. Crocker</td>
</tr>
<tr>
<td>RFC 698</td>
<td>Telnet extended ASCII option T. Mock</td>
</tr>
<tr>
<td>RFC 726</td>
<td>Remote Controlled Transmission and Echoing Telnet option J. Postel, D. Crocker</td>
</tr>
<tr>
<td>RFC 727</td>
<td>Telnet logout option M.R. Crispin</td>
</tr>
<tr>
<td>RFC 732</td>
<td>Telnet Data Entry Terminal option J.D. Day</td>
</tr>
<tr>
<td>RFC 733</td>
<td>Standard for the format of ARPA network text messages D. Crocker, J. Vittal, K.T. Pogran, D.A. Henderson</td>
</tr>
<tr>
<td>RFC 734</td>
<td>SUPDUP Protocol M.R. Crispin</td>
</tr>
<tr>
<td>RFC 735</td>
<td>Revised Telnet byte macro option D. Crocker, R.H. Gumpertz</td>
</tr>
<tr>
<td>RFC 736</td>
<td>Telnet SUPDUP option M.R. Crispin</td>
</tr>
<tr>
<td>RFC 749</td>
<td>Telnet SUPDUP—Output option B. Greenberg</td>
</tr>
<tr>
<td>RFC 765</td>
<td>File Transfer Protocol specification J. Postel</td>
</tr>
</tbody>
</table>
RFC 768
User Datagram Protocol J. Postel

RFC 779
Telnet send-location option E. Killian

RFC 791
Internet Protocol J. Postel

RFC 792
Internet Control Message Protocol J. Postel

RFC 793
Transmission Control Protocol J. Postel

RFC 820
Assigned numbers J. Postel

RFC 823
DARPA Internet gateway R. Hinden, A. Sheltzer

RFC 826
Ethernet Address Resolution Protocol: Or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware D. Plummer

RFC 854
Telnet Protocol Specification J. Postel, J. Reynolds

RFC 855
Telnet Option Specification J. Postel, J. Reynolds

RFC 856
Telnet Binary Transmission J. Postel, J. Reynolds

RFC 857
Telnet Echo Option J. Postel, J. Reynolds

RFC 858
Telnet Suppress Go Ahead Option J. Postel, J. Reynolds

RFC 859
Telnet Status Option J. Postel, J. Reynolds

RFC 860
Telnet Timing Mark Option J. Postel, J. Reynolds

RFC 861
Telnet Extended Options: List Option J. Postel, J. Reynolds

RFC 862
Echo Protocol J. Postel

RFC 863
Discard Protocol J. Postel

RFC 864
Character Generator Protocol J. Postel

RFC 865
Quote of the Day Protocol J. Postel

RFC 868
Time Protocol J. Postel, K. Harrenstien

RFC 877
Standard for the transmission of IP datagrams over public data networks J.T. Korb

RFC 883
Domain names: Implementation specification P.V. Mockapetris

RFC 884
Telnet terminal type option M. Solomon, E. Wimmers
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
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
RFC 1032
Domain administrators guide M. Stahl

RFC 1033
Domain administrators operations guide M. Lottor

RFC 1034
Domain names—concepts and facilities P.V. Mockapetris

RFC 1035
Domain names—implementation and specification P.V. Mockapetris

RFC 1038
Draft revised IP security option M. St. Johns

RFC 1041
Telnet 3270 regime option Y. Rekhter

RFC 1042
Standard for the transmission of IP datagrams over IEEE 802 networks J. Postel, J. Reynolds

RFC 1043
Telnet Data Entry Terminal option: DODIIS implementation A. Yasuda, T. Thompson

RFC 1044
Internet Protocol on Network System's HYPERchannel: Protocol specification K. Hardwick, J. Lekashman

RFC 1053
Telnet X.3 PAD option S. Levy, T. Jacobson

RFC 1055
Nonstandard for transmission of IP datagrams over serial lines: SLIP J. Romkey

RFC 1057

RFC 1058
Routing Information Protocol C. Hedrick

RFC 1060
Assigned numbers J. Reynolds, J. Postel

RFC 1067

RFC 1071
Computing the Internet checksum R.T. Braden, D.A. Borman, C. Partridge

RFC 1072
TCP extensions for long-delay paths V. Jacobson, R.T. Braden

RFC 1073
Telnet window size option D. Waitzman

RFC 1079
Telnet terminal speed option C. Hedrick

RFC 1085
ISO presentation services on top of TCP/IP based internets M.T. Rose

RFC 1091
Telnet terminal-type option J. VanBokkelen

RFC 1094
NFS: Network File System Protocol specification Sun Microsystems

RFC 1096
Telnet X display location option G. Marcy

RFC 1101
DNS encoding of network names and other types P. Mockapetris
RFC 1112
Host extensions for IP multicasting S.E. Deering

RFC 1113
Privacy enhancement for Internet electronic mail: Part I — message encipherment and authentication procedures J. Linn

RFC 1118
Hitchhiker’s Guide to the Internet E. Krol

RFC 1122
Requirements for Internet Hosts—Communication Layers R. Braden, Ed.

RFC 1123
Requirements for Internet Hosts—Application and Support R. Braden, Ed.

RFC 1146
TCP alternate checksum options J. Zweig, C. Partridge

RFC 1155
Structure and identification of management information for TCP/IP-based internets M. Rose, K. McCloghrie

RFC 1156
Management Information Base for network management of TCP/IP-based internets K. McCloghrie, M. Rose

RFC 1157

RFC 1158
Management Information Base for network management of TCP/IP-based internets: MIB-II M. Rose

RFC 1166
Internet numbers S. Kirkpatrick, M.K. Stahl, M. Recker

RFC 1179
Line printer daemon protocol L. McLaughlin

RFC 1180
TCP/IP tutorial T. Socolofsky, C. Kale

RFC 1183
New DNS RR Definitions C.F. Everhart, L.A. Mamakos, R. Ullmann, P.V. Mockapetris

RFC 1184
Telnet Linemode Option D. Borman

RFC 1186
MD4 Message Digest Algorithm R.L. Rivest

RFC 1187
Bulk Table Retrieval with the SNMP M. Rose, K. McCloghrie, J. Davin

RFC 1188
Proposed Standard for the Transmission of IP Datagrams over FDDI Networks D. Katz

RFC 1190
Experimental Internet Stream Protocol: Version 2 (ST-II) C. Topolcic

RFC 1191
Path MTU discovery J. Mogul, S. Deering

RFC 1198
FYI on the X window system R. Scheifler

RFC 1207
FYI on Questions and Answers: Answers to commonly asked “experienced Internet user” questions G. Malkin, A. Marine, J. Reynolds

RFC 1208
Glossary of networking terms O. Jacobsen, D. Lynch

Related protocol specifications 1059
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
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 1061
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 Juhu 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
Related protocol specifications 1063
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 Notifications K. 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 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 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 1069
RFC 2640
Internationalization of the File Transfer Protocol B. Curtin

RFC 2665
Definitions of Managed Objects for the Ethernet-like Interface Types J. Flick, J. Johnson

RFC 2671
Extension Mechanisms for DNS (EDNS0) P. Vixie

RFC 2672
Non-Terminal DNS Name Redirection M. Crawford

RFC 2675
IPv6 Jumbograms D. Borman, S. Deering, R. Hinden

RFC 2710
Multicast Listener Discovery (MLD) for IPv6 S. Deering, W. Fenner, B. Haberman

RFC 2711
IPv6 Router Alert Option C. Partridge, A. Jackson

RFC 2740
OSPF for IPv6 R. Coltun, D. Ferguson, J. Moy

RFC 2753
A Framework for Policy-based Admission Control R. Yavatkar, D. Pendarakis, R. Guerin

RFC 2782
A DNS RR for specifying the location of services (DNS SRV) A. Gubrandsen, P. Vixix, L. Esibov

RFC 2821
Simple Mail Transfer Protocol J. Klensin, Ed.

RFC 2822
Internet Message Format P. Resnick, Ed.

RFC 2840
TELNET KERMIT OPTION J. Altman, F. da Cruz

RFC 2845
Secret Key Transaction Authentication for DNS (TSIG) P. Vixie, O. Gudmundsson, D. Eastlake 3rd, B. Wellington

RFC 2851
Textual Conventions for Internet Network Addresses M. Daniele, B. Haberman, S. Routhier, J. Schoenwaelder

RFC 2852
Deliver By SMTP Service Extension D. Newman

RFC 2874
DNS Extensions to Support IPv6 Address Aggregation and Renumbering M. Crawford, C. Huitema

RFC 2915
The Naming Authority Pointer (NAPTR) DNS Resource Record M. Mealling, R. Daniel

RFC 2920
SMTP Service Extension for Command Pipelining N. Freed

RFC 2930
Secret Key Establishment for DNS (TKEY RR) D. Eastlake, 3rd

RFC 2941
Telnet Authentication Option T. Ts’o, ed., J. Altman

RFC 2942
Telnet Authentication: Kerberos Version 5 T. Ts’o

RFC 2946
Telnet Data Encryption Option T. Ts’o

RFC 2952
Telnet Encryption: DES 64 bit Cipher Feedback T. Ts’o
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
RFC 3417

RFC 3418

RFC 3419
Textual Conventions for Transport Addresses M. Daniele, J. Schoenwaelder

RFC 3484
Default Address Selection for Internet Protocol version 6 (IPv6) R. Draves

RFC 3493

RFC 3513
Internet Protocol Version 6 (IPv6) Addressing Architecture R. Hinden, S. Deering

RFC 3526
More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE) T. Kivinen, M. Kojo

RFC 3542
Advanced Sockets Application Programming Interface (API) for IPv6 W. Richard Stevens, M. Thomas, E. Nordmark, T. Jinmei

RFC 3566
The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec S. Frankel, H. Herbert

RFC 3569
An Overview of Source-Specific Multicast (SSM) S. Bhattacharyya, Ed.

RFC 3584

RFC 3602
The AES-CBC Cipher Algorithm and Its Use with IPsec S. Frankel, R. Glenn, S. Kelly

RFC 3629
UTF-8, a transformation format of ISO 10646 R. Kermode, C. Vicisano

RFC 3658
Delegation Signer (DS) Resource Record (RR) O. Gudmundsson

RFC 3678
Socket Interface Extensions for Multicast Source Filters D. Thaler, B. Fenner, B. Quinn

RFC 3715
IPsec-Network Address Translation (NAT) Compatibility Requirements B. Aboba, W. Dixon

RFC 3810

RFC 3826

RFC 3947
Negotiation of NAT-Traversal in the IKE T. Kivinen, B. Swander, A. Huttunen, V. Volpe

RFC 3948
UDP Encapsulation of IPsec ESP Packets A. Huttunen, B. Swander, V. Volpe, L. DiBurro, M. Stenberg

RFC 4001
Textual Conventions for Internet Network Addresses M. Daniele, B. Haberman, S. Routhier, J. Schoenwaelder

RFC 4007
IPv6 Scoped Address Architecture S. Deering, B. Haberman, T. Jinmei, E. Nordmark, B. Zill
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.
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.
Bibliography

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>

© Copyright IBM Corp. 2000, 2019
<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
</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

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: IP User's Guide and Commands</td>
<td>SC27-3662</td>
<td>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.</td>
</tr>
<tr>
<td>z/OS Communications Server: IP System Administrator's Commands</td>
<td>SC27-3661</td>
<td>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.</td>
</tr>
<tr>
<td>z/OS Communications Server: SNA Operation</td>
<td>SC27-3673</td>
<td>This document serves as a reference for programmers and operators requiring detailed information about specific operator commands.</td>
</tr>
<tr>
<td>z/OS Communications Server: Quick Reference</td>
<td>SC27-3665</td>
<td>This document contains essential information about SNA and IP commands.</td>
</tr>
</tbody>
</table>
## Customization

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: SNA Customization</td>
<td>SC27-3666</td>
<td>This document enables you to customize SNA, and includes the following information:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Communication network management (CNM) routing table</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Logon-interpret routine requirements</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Logon manager installation-wide exit routine for the CLU search exit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• TSO/SNA installation-wide exit routines</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SNA installation-wide exit routines</td>
</tr>
</tbody>
</table>

## 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>
<tr>
<td>Title</td>
<td>Number</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------------------</td>
<td>-----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>z/OS Communications Server: SNA Programmer’s LU 6.2 Reference</td>
<td>SC27-3670</td>
<td>This document provides reference material for the SNA LU 6.2 programming interface for host application programs.</td>
</tr>
<tr>
<td>z/OS Communications Server: CSM Guide</td>
<td>SC27-3647</td>
<td>This document describes how applications use the communications storage manager.</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: 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>
<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: 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>
</tbody>
</table>

**Messages and codes**

<table>
<thead>
<tr>
<th>Title</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>z/OS Communications Server: SNA Messages</td>
<td>SC27-3671</td>
<td>This document describes the ELM, IKT, IST, IUT, IVT, and USS messages. Other information in this document includes:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Command and RU types in SNA messages</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Node and ID types in SNA messages</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Supplemental message-related information</td>
</tr>
<tr>
<td>Title</td>
<td>Number</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------------</td>
<td>-----------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<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.