Release Notes

IBM® Tivoli® Identity Manager

Active Directory (WinAD) Adapter

Version 5.0.14

First Edition (February 10, 2012)

This edition applies to version 5.0 of Tivoli Identity Manager and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Preface

Welcome to the IBM Tivoli Identity Manager Active Directory (WinAD) Adapter.

These Release Notes contain information for the following products that was not available when the IBM Tivoli Identity Manager manuals were printed:

- IBM Tivoli Identity Manager Active Directory Adapter Installation and Configuration Guide

Adapter Features and Purpose

The Active Directory Adapter is designed to create and manage accounts on Microsoft Active Directory. The adapter runs in “agentless” mode and communicates using Microsoft ADSI API and CDOEXM (for exchange communication) to the systems being managed.

IBM recommends the installation of this adapter in “agentless” mode on a computer in the domain being managed. Installation on a Domain Controller is not recommended. A single copy of the adapter can handle multiple Identity Manager Services. The deployment configuration is based, in part, on the topology of your network domain, but the primary factor is the planned structure of your Identity Manager Provisioning Policies and Approval Workflow process. Please refer to the Identity Manager Information Center for a discussion of these topics.

The Identity Manager adapters are powerful tools that require Administrator Level authority. Adapters operate much like a human system administrator, creating accounts, permissions and home directories. Operations requested from the Identity Manager server will fail if the adapter is not given sufficient authority to perform the requested task. IBM recommends that this adapter run with administrative (root) permissions.

Contents of this Release

Adapter Version

Component	Version
Release Date	February 10, 2012
Adapter Version	5.0.14
Component Versions	Adapter Build 5.0.1024 Profile 5.0.1011 ADK 5.20
Documentation	Active Directory Adapter Installation and Configuration Guide SC23-6175-00 Active Directory Adapter User Guide SC23-6176-00 Password Synchronization for Active Directory Plug-in Installation and Configuration Guide SC23-6178-00

New Features

Enhancement # (FITS)	Description
	Items included in current release
MR0721117156	WinAD/WinAD64 adapter: Allow support for Octet String data type in extended attributes. The adapter now supports Octet String as an extended attribute type. It is assumed to be passed as a string value with an even number of hexadecimal characters.
MR0629114418	WinAD: Unable to provision WinAD Extended attributes without MS Exchange Management Tools installed locally. The adapter now treats the extended exchange attributes as account attriubutes since they are set via LDAP calls anyway and do not require an exchange client.
NA	Enhancements to adapter based event notification. Now searches global catalog for domain controllers when scanning event logs for group membership changes. The actual group membership change event is sent instead of returning the full list of groups for the user account. Supports all configured EN contexts with one pass of event log scanning per cycle instead of scanning the event logs for each EN context.
	Items included in 5.0.13 release
	None
	Items included in 5.0.12 release
	None
	Items included in 5.0.11 release
MR0204103013	AD adapter extended schema support for data type DNWithBinary. See additional information in the “Configuration Notes” section.
	Items included in 5.0.10 release
MR0302105547	Use erADLastLogonTimeStamp for AD dormant account report
MR0210102732	OCS support for AD adapter.
MR0205101253	Windows 2008 R2 Core support for AD password sync plug-in and Adapter.
MR0226101912	WinAD: Need a way to configure the length of the wait period for the retries of Win AD 64-bit Adapter for reconciliation
MR0501091918	ADK changes to DAML for adding timeout on read.

MR0501091927	Incorrect error message during an AD account modification.
MR0218091930	Unlock WinAD account without password reset.
MR0830071536	Customer wants ability to add/modify userCannotChangePassword ACE without having to use the Domain Administrator account for the AD adapter service.
	Items included in 5.0.9 release
	None
	Items included in 5.0.8 release
N/A	Add support for Windows 2008 R2 as a target AD. Add support for Windows 2008 R2 as a host platform for the adapter.
MR052609514	Customer want to use the extended attribute transformed name on the ITIM side using the AD 5.0.5 adapter on ITIM 4.6 Server.

Enhancement # (FITS)	Description
	Items included in 5.0.7 release
N/A	Added Registry Key “SearchTimeout” to work around Microsoft hang in AD LDAP client present in Windows 2003. Additional information can be found in the “Configuration Notes” section.
N/A	Enhanced form validation. Added constraint to exclude special characters forbidden by Active Directory in the samAccountName. Additional information can be found in the “Configuration Notes” section.
MR050109660	Added retry on mailbox creation to avoid ADSI error 0x80004005 on slower networks.
	Items included in 5.0.6 release
MR0212084734	Enhancement to handle failover of target systems by supporting multiple target servers in basepoint.
MR032609161	WinAD Enhancement to handle erADEProxyAddresses attribute in add/delete as well as in replace format.
N/A	Improve primary group lookup using cache.
	Items included in 5.0.5 release
OSDB	Added support for Windows 2008
MR0831085255	Add support for management of field msRADIUSFramedIPAddress in Active Directory.
MR0808084336	Add support for management of attribute msexchrequireauthtosendto.
	Items included in 5.0.4 release
MR0825086415	Enhance the adapter to allow an account to be changed from a mail-user to a to mailbox user in a single transaction.
MR121707567	Enhance the adapter to support the DL email attribute.
N/A	Add support for Windows 2008 as a target AD. Add support for Windows 2008 as a host platform for the adapter.

Enhancement # (FITS)	Description
	Items included in 5.0.3 release
MR0427076459	Add support for lastLogonTimeStamp attribute. See additional information in the “New Features of this Adapter” section.
MR1118053029	WinAD: Enhance the adapter to handle all the options of the "Remote Access Permission" property NOTE: If you have any business logic (e.g. Provisioning Policy Parameter) around attribute "erADAllowDialin" then you must modify it to use "erADExDialin". See additional information in the “New Features of this Adapter” section.
	Items included in 5.0.2 release
N/A	Adapter has been enhanced to support Group DN and Group GUID values. See Configuration Notes section for additional information.

Closed Issues

Internal#	APAR#	PMR# / Description
		Items closed current version
	IV11173	False warnings generated during adagent home directory processing. The adapter was marking attributes as failed when unable to access the home directory share even though they were not included in the adapter resulting in warning. Error is now only set if the attribute existed in the request
	IV13282	Unable to provision account when country code is Serbia. The country code list was updated to the most recent. The account form and customlables files in the profile were also updated to add the new countries to the list.
		Items closed in 5.0.13 version
	IZ98592	0738,999,616 Multiple values returned for erADEAllowPermTo1Level during recon. FIX: The adapter will now only returns the first Access Control Entry (ACE) for "SELF". If multiple ACEs exist for SELF, the erADEAllowPermTo1Level is only returned for the first ACE.
	IZ94371	77770,550,000 If the erPassword attribute value cannot be set, it will be displayed in clear text in the adapter log file. FIX: The adapter has been modified so that any attribute whose name contains "pwd" or "pass" will now only show "*****" as the value in the unmodified attributes list.
	IZ89492	85059,999,724 ADK adapter message :starting new SSL connection thread” FIX: The text "SSL" has now been removed from this message. The IO library will clearly log the ssl handshake before this message is logged. It is not necessary to state that the connection is SSL or not
		Items closed in 5.0.12 version
	IZ89623	CO attribute doesn't get the correct value of the country name.
	IZ91338	Windows ad adapter crashes when searching for duplicate UPNs.
		Items closed in 5.0.11 version
		None
		Items closed in 5.0.10 version
	N/A	33834,999,760 WinADAdapter failed to load exschema.txt at system bootup.
	IZ58983	94774,227,000 CERTTOOL.EXE UNABLE TO LIST CERTIFICATE VERISIGN CLASS 3 SECURE SERVER CA - G2.

36607		N/A ADK incorrect return status for multivalued attr with spl chars.
		Items closed in 5.0.9 version
		None
		Items closed in 5.0.8 version
		None
		Items closed in 5.0.7 version
	IZ73004	12546,999,678 WinAD adapter is not setting failure for RAS related attributes.
37127		WinAD adapter incorrectly identifies status of RUS service.

INTERNAL#	APAR#	PMR# / Description
		Items closed in 5.0.6 version
	IZ 70467	24803,422,000 AD agent crashes when modify request includes changing erADIsAccountLocked to a space.
	IZ63150	58172,033,724 TSM backup fails on (32 bit) Windows if ITIM adapters are installed.
	IZ64602	75002,442,000 The modify operation fail only for MailboxStore
	IZ65637	39372,180,000 Custom “UTC Coded Time” attribute value not correctly returned from AD adapter during recon.
	IZ66086	07172,035,724 Reconciliation error with WinAD 64-bit adapter 5.0.8.
	IZ67106	79658,442,000 Intermittent 0x80004002 error when attempting to manage/provision mailboxes.
	IZ60509	36047,660,706 Changes to WTS home dir and drive letter no committed to AD.
36620		N/A Problem with WTS Boolean attributes during recon.
36208		N/A WinAD adapter returns incorrect container when CN contains “=”.
	N/A	81172,379,000 RUS and EAG errors during WinAD Adapter Add request. Following changes are made in adapter logging. • Adapter will log "Domain Flat Name" in Adapter log file. • Adapter log messages are enhanced to provide more error message with the error code when searching for RAS Server Name.

INTERNAL#	APAR#	PMR# / Description
	IZ45782	92561,077,724 Checkboxes in account form for exchange cause transaction "warning"
	IZ43288	66187,379,000 WinAD adapter CN attribute is incompatible with TIM Alert Processing features.
	IZ52976	06524,6X1,760 WinAD group membership modifications fail if group contains "#" and the adapter is set to use CN for group linkage.
	IZ51132	19725,379,000 Group names containing a "/" character are not properly parsed if the adapter is set to use DN group linkage.
	IZ54007	06743,6X1,760 Adapter fails to add group membership of the group contains an unmatched closing parentheses ")" character.

	N/A	34450,057,649 Added a “SearchTimeout” option to work around AD hang on LDAP read in Windows 2003. See “Configuration Notes” section for more information.
	N/A	29147,227,000 Adapter may crash if an inactive user is moved to a new OU.
36159	N/A	N/A Adapter may not return an error message if a delete user request if the base point bind fails.
36160	N/A	N/A Correct issue with filter substitution of erACContainer that may occur when the “Adapter Filtering” option is used with TIM v4.6,
	IZ47418	10132,922,848 WinAD adapter crashes randomly. Fixed in ADK by adding Read Timeout to avoid firewall time out issues.
	IZ49418	83618,550,000 Event Notification displays incorrect value for next run time after “Run Event Now” option is selected.
	IZ52909	39888,227,000 PERFORMING FILTERED RECON WITH ITIM V4.6 SERVER RETURNS CONTAINER OBJECTS CAUSING RECON ERRORS IN ITIM.
	IZ53580	78434,035,649 ERROR MESSAGE CTGIMD014I FOR AD GROUPS WHEN PERFORMING AD RECON WITH ITIM 4.6 PROFILE.
36212		N/A WinAD Adapter fails to find user when eruid contains (or).
36238		N/A WinAD Adapter returns incorrect error for duplicate UPN.
36240		N/A WinAD Adapter terminates when an error occurs in reconciliation.

INTERNAL#	APAR#	PMR# / Description
		Items closed in 5.0.6 version
	IZ38367	59070,082,000 Two users get created in AD and UPN without checking.
	IZ44751	57426,004,000 WinAD64 Crashes on user ADD, with no Basepoint.
	IZ42815	67170,650,706 AD Adapter cannot be installed under "C:\Program files\" Directory.
	N/A	N/A WinAD Adapter Crashes Randomly (related to IZ47418) ADK crash in SSL_read_error while reading error message after a handshake failure.
	IZ47221	83137,379,000 64 Bit AD Adapter not handling multi-line-attribute values properly.
	IZ45782	92561,077,724 Check box in form for exchange attribute causes error in agent log
35175	N/A	N/A Adapter returns unreadable error message for multivalued attribute.
35017	N/A	N/A ADK Crash in recon request when user EntryDN contain 'pass'/'pwd'.
34467	N/A	N/A Adapter to ignore exchange attrs after mail status is cleared.
33974	N/A	N/A WinAD Adapter recon returning unwanted exchange attributes.

INTERNAL#	APAR#	PMR# / Description
		Items closed in 5.0.5 version
	N/A	Updated the PW Sync Plug-In 1) MR1017081822 - create a version of PW Sync without 3rd party dependencies 2) PMR 37512,379,000 - Added more logging in domain bind and user lookup. 3) 77327,021,724 - Code changes for ignoring TIM response when TIM application is down Down
	IZ35932	16714,7TD,000 AD Adapter recon fails because the recon event crashes the service when parsing security descriptor data for a user it recons. A new registry key 'ReconMailboxPermissions' is introduced to provide a workaround for this issue. This registry key is set to TRUE by default and will reconcile mailbox security permissions attributes. Setting registry key 'ReconMailboxPermissions' to FALSE will also improve reconciliation performance.
33671	N/A	N/A Attribs erADERstrctAdrsFg & erADERstrctAdrsLs are not reconciled.
		Items closed in 5.0.4 version
33183	N/A	N/A Agent doesn't retrieve correct value of exchange attribute 'Permanent delete only after backup' through recon.
	IZ31246	78773,000,738 Recon fails if base-point contains Chinese character.
		Items closed in 5.0.3 version
	IZ24750	42485,999,866 AD Password Sync plug-in 5.0.1 does not check the origin of password change. NOTE: Password Change recursion control has moved from the adapter to the Identity Manager server to provide improved support for high-availability configurations. See "Password Sync Recursion Control" in the New Features section of this document.
	IZ24107	37721,550,000 WinAD Adapter returns success for add request with home directory creation, but the home directory is not created on the resource.

INTERNAL#	APAR#	PMR# / Description
	IZ13159	19535,005,000 Add support for the third dial-in option "Control access through Remote Access Policy"
32557 32558 32579	N/A	N/A CMVC #32557 - AD Agent sets WTS drive letter instead of WTS Home directory in "Local path" on AD. CMVC #32558 - WinAD Adapter returns success but cn is not set on AD. Attributes mobile, postofficebox, givenname, sn, telephonenumber, street, pager, l, mail, homephone, postalcode, title, description and st are send as add-delete for a modify request instead of replace. CMVC #32579 - WinAD worngly sets Dialin Callback option as "Set by caller (Routing and Remote Service only)" when "Fixed callback number" is selected.
	IZ25019	65296,001,822 Problem with setting the 'erlogontimes' attribute to 'Mid-Hour'. The adapter throws an error expecting a 168 bit string. NOTE: AD Adapter erLogonTimes can only be set to hourly and not to 'Mid-Hour'.
		Items closed in 5.0.2 version
	IZ09072	20243,999,760 Unable to set some AD account attributes that are related to passwords (TIM Express only).
	IZ14154	38318,L6Q,000 Incorrect response on group modification. Status code from the adapter may report “error” or “warning” if the only attribute value set is a group.
	IZ14468	23949,292,848 The street attribute is not set properly on the resource, the XML parser in ADK removes \r from newline.

Known Issues

INTERNAL#	APAR#	PMR# / Description
N/A	N/A	**Editing adapter profiles on UNIX or Linux.** The adapter profile JAR file may contain ASCII files created using MS-DOS ASCII format (i.e. schema.dsml, CustomLabels.properties, and service.def). If you edit a MS-DOS ASCII file in Unix you will often see the characters ^M at the end of each line. This is the extra character 0x0d that is used to indicate a new line of text in MS-DOS. There are tools, such as dos2unix, that can be used to strip out the ^M character. In addition, there are text editors that will ignore the ^M character. If you are using the vi editor, you can strip out the ^M character as follow: From the vi's command mode: :%s/^M//g followed by pressing Enter. The ^M (or Ctrl-M) typed to show it here should actually be entered by pressing ^v^M in sequence. (The ^v preface tells vi to use the next keystroke literally instead of taking it as a command.)
N/A	N/A	**Using the Upgrade Option:** The Upgrade option is applicable only to 5.0.x maintenance upgrades. The upgrade option is not designed for v4.6 to v5.0 migrations. NOTE: After using "Update Installation" option with higher version of Adapter, an extra folder with name "_uninst2" is created. It can be ignored. To Uninstall the Adapter, use "_uninst" folder.

INTERNAL#	APAR#	PMR# / Description
N/A	N/A	PMR 83403,057,649 - AD Agent hangs on second error Work around implemented for Microsoft issue (KB article 293278). For user add request, the agent binds to the basepoint/default domain and checks to see if the specified user already exists. Each operation is run in a separate thread which first initializes COM using CoInitialize and upon operation completion calls CoUninitialize to uninitialize COM. However for second request, the connection is reused and agent quickly establishes connection with AD. MS KB article 293278 (http://support.microsoft.com/default.aspx?scid=kb;en-us;293278) states that 'PRB: Problems When You Call CoInitialize and CoUninitialize Repeatedly in Multithreaded Apartment' - To avoid the issue, a small delay (1 sec) has been inserted before COM is uninitialized at the end of operation. This delay is ONLY applicable when 'User already exists' condition is encountered and DOES NOT affect any other functionality.
		Below is list of TIM AD adapter attributes and corresponding attribute in Active Directory erADLastFailedLogin -> badPasswordTime erADBadLoginCount -> badPwdCount erADLastLogoff -> lastLogoff erADLastLogon -> lastLogon The above attributes are nonreplicated attribute, which means that each domain controller holds its own copy of the attribute, likely with different values. The WinAD adapter may not show actual value for these attributes.
N/A	N/A	**Group Names with Tilde (“~“) Character** This known issue is only applicable to IBM Tivoli Identity Manger v4.6. This is when a group’s CN contains the character ‘~’ and you select and add that group to the Groups attribute on account form. When the request is submitted to the adapter, characters before the ‘~’ characters are only added to the erGroup attribute. IBM Tivoli Identity Manager treats the ‘~’ character as delimiter and truncates the characters after it in the group name. For example, If you add a group whose CN is TEST~TST to a user and submit the account form, in the request only TEST will be added as value for erGroup attribute and not TEST~TST. **Workaround**: Avoid using this special character ‘~’ in group name on Active Directory.

INTERNAL#	APAR#	PMR# / Description
N/A	N/A	**ERUID does not allow special characters** This known issue applies to IBM Tivoli Identity Manger (ITIM) v4.6 & v5.0. When a user account's samAccountName attribute on Active Directory contains characters like '(' and/or ')' and a Full Reconciliation is performed. The Full Reconciliation will complete with a warning. Please note that the samAccountName attribute of a user account on Active Directory maps to the eruid attribute on ITIIM. For example, If you have user accounts on Active directory with following samAccountName TestUser(_01 TestUser)_02 (TestUser _04(TestUser _05) The Full Reconciliation will result in warning with following error message: CTGIMD014I 7 reconciliation entries were not processed for the following entries: eruid=TestUser(_01; eruid= TestUser)_02; eruid=(TestUser _04(; eruid=) TestUser _05);. **Workaround**: Avoid use of special characters like '(' or ')' for the samAccountName attribute of user account on Active Directory.
		Known Issues for Adapter based Event Notification When you modify the 'User logon name (pre-Windows 200)' of a user account on Active Directory, Adapter based event notification results in the creation of two accounts on IBM Tivoli Identity Manager. One of the accounts is created using the old name and other account with modified name (Orphan account). **Workaround:** As far as possible avoid renaming user accounts on the Active Directory directly. Perform user modifications only through Tivoli Identity Manager. However if you rename a user account on Active Directory, perform full reconciliation to avoid creation of duplicate user accounts on Tivoli Identity Manager. When you configure Active Directory for not caching the deleted objects in the deleted objects container, then the adapter cannot track the deletion of an object (user/group/organizational unit) on the Active Directory. As a result IBM Tivoli Identity Manager will not be updated for the deleted objects. **Workaround**: no workaround. When you remove an organization unit (container) on Active Directory which contains user accounts, then all the user accounts in the organization unit are removed from Active Directory. The user accounts which are removed as a result of deletion of the organization unit are not traced by the adapter based event notification. The information of such user accounts will not get updated on IBM Tivoli Identity Manager.

		Workaround: Before you remove a container on the Active Directory, remove all the user accounts from this container and from the sub-containers. This way the adapter based event notification will be able to trace user account deletions. An event notification context maintains each object (user, group, container, and mailstore) by generating a key in a local database. The maximum key length of the database is 64 characters. When you have containers, groups, and mailstores in Active Directory with name containing more than 64 characters, the adapter based event notification may function incorrectly. The event notification might add duplicate entries in database or read incorrect entry from the database. This may result in incorrect updation to objects in IBM Tivoli Identity Manager. **Workaround**: When you create a container, group, or mailstore on Active Directory select a name such that the first 50 characters don't match with the first 50 characters in the name of other objects. Before removing a group from Active Directory, remove all the members from the group. When you do so all the member objects are updated properly on IBM Tivoli Identity Manager. **Workaround**: Before deleting a group on the resource, empty it's 'Members', i.e. remove the objects which are members of this group. This will ensure that the dependant Account's gets properly updated in IBM Tivoli Identity Manager (ITIM) server When the log information in the event viewer log file is cleared, then during the next event notification operation you may get the following error in Adapter log file. *'Unable to read the last highest record from the log. Error code: 0x00000057 - The parameter is incorrect*'. **Workaround**: No action to be taken. The next adapter based event notification will succeed to read the event log properly. When a user account is moved from a container which is either the base point or a sub-container of the base point to a container which is not in the base point, then after running event notification this user account will not get updated on IBM Tivoli Identity Manager. Ideally this user no longer belongs the corresponding service on IBM Tivoli Identity Manager, as it is moved to a container which is not in the base point of this service. **Workaround**: When you move a user account on Active Directory to a container which is not under the base point, then perform a full reconciliation to update the user accounts on IBM Tivoli Identity Manager.

INTERNAL#	APAR#	PMR# / Description
		Issue with adapter based filtering Avoid use of cn attribute of the common schema in a filter with objectclass value as erADContainer. Although the filter is perfectly valid for Active Directory and IBM Tivoli Identity Manager will not give any error while evaluating and submitting the filter to adapter. However after the adapter processes the filter and returns matching entries, IBM Tivoli Identity Manager will give failure while processing the returned entries. This is because the CN attribute does not belong to erADContainer class on IBM Tivoli Identity Manager Schema for Active Directory profile. For example, The following filter will result in failure for the reconciliation request on IBM Tivoli Identity Manager (&(objectclass=erADContainer)(cn=MyContainer))
		Class 3 Certificate Installation Class 3 Certificates (class 3 secure server CA-G2) are not written properly to "DamlCACerts.pem" file through CertTool.exe Utility. The certificate data is written twice between BEGIN CERTIFICATE and END CERTIFICATE. **Work around:** To correct this issue, please follow the below steps and edit "DamlCACerts.pem" file present in "<Adapter installation path>\data" folder. **Step 1.** Start the CertTool utility **Step 2.** Import the class 3 CA certificate by using "F" option from the main menu of CertTool Utility. **Step 3.** Once the class 3 CA certificate is successfully installed, open "DamlCACerts.pem" file stored in the "<Adapter installed path>\data" folder using text editor. **Step 4**. Delete the class 3 CA certificate data (i.e. content between BEGIN CERTIFICATE and END CERTIFICATE) from "DamlCACerts.pem". **Step 5**. Open class 3 CA certificate file using text editor and copy the certificate data (between the BEGIN CERTIFICATE and END CERTIFICATE) **Step 6**. Paste the certificate data to "DamlCACerts.pem" file between the BEGIN CERTIFICATE and END CERTIFICATE lines of same class 3 CA Certificate. If more than one class 3 certificates are installed then you can identify the certificate using issuer and subject data. **Step 7.** Save "DamlCACerts.pem" file. **Step 8.** To verify the "DamlCACerts.pem" file is edited properly, display certificate information by using option "E" from the main menu of CertTool Utility. **Note:** Please note that this issue is seen after installing class 3 CA certificate. If you correct the DamlCACerts.pem and then install another class 3 CA certificate, the newly installed class 3 CA certificate will show same issue. This issue is also seen when you delete any certificate using option "G" from the main menu of CertTool

		utility. The delete option will affect all remaining class 3 CA certificate and you have to follow step 1 to 8 to correct the DamlCACerts.pem file.
		1. **NetBIOS name must not be greater than 15 characters**: The NetBIOS name is 16 ASCII characters, however Microsoft limits the NetBIOS name to 15 characters and reserves the 16th character as a NetBIOS Suffix. The Windows Active Directory adapter does utilize a NetBIOS based lookup for finding the RAS Server and WTS Server. Adapter is limited by the Windows restriction of having a 15 character maximum value for a NetBIOS computer name\NetBIOS domain name. If the NetBIOS domain name or NETBIOS computer name is more then 15 characters then Windows Active directory adapter will display following error message for RAS Server lookup. ERROR_INVALID_DOMAINNAME 1212 "The format of the specified domain name is invalid." Please refer the Microsoft link related to Naming conventions where it specifies a maximum name length of 15 characters: http://support.microsoft.com/kb/909264 **Workaround:** Set the adapter registry key "ForceTerminalServerLookup" and "ForceRASServerLookup" to FALSE value using agentCfg utility provided by adapter. Specify one or more then one target servers for the base point on the Active Directory Adapter service form on IBM Tivoli Identity Manager Server. Specify the target servers which are configured as RAS Server or WTS server. This will resolve RAS and WTS Server lookup issue for Error Code: 1212. Each target server must be separated by '|' (Pipe character) For example: Base Point DN on the service form with only one target server: DC01/OU=engineering,DC=irvine,DC=IBM,DC=com Base Point DN on the service form with more than one target server: DC01|DC02|DC03/OU=engineering,DC=irvine,DC=IBM,DC=com For more detail on configuring user basepoint, please refer the section "Configuring the Users Base Point for the adapter" in Active Directory Adapter Installation and Configuration Guide.

<table>
<tr><th>INTERNAL#</th><th>APAR#</th><th>PMR# / Description</th></tr>
<tr><td></td><td></td><td>
Issue with Special Characters during Recon Operation:

Reconciliation will fail on Tivoli Identity Manager if an attribute’s value in recon entry contains one or more of the special characters listed in the below table which are not transformed to their equivalent XML format.
When ADK reads an attribute’s value it searches for each of the XML transformed value, as listed in below table, in the value string. If it finds any one of these then it considers that entire string is already transformed and will not perform any transformation on that string. In this case ADK does not attempt to check if there are any untransformed special characters in the string. If the value contains other untransformed special characters then they are not transformed by ADK. When Tivoli Identity Manager processes such recon entries they will be failed.

<table>
<tr><td>Special Character</td><td>Equivalent XML transformation</td></tr>
<tr><td>& (ampersand)</td><td>&</td></tr>
<tr><td>' (apostrophe or single quote)</td><td>'</td></tr>
<tr><td>" (double-quote)</td><td>"</td></tr>
<tr><td>< (less-than)</td><td><</td></tr>
<tr><td>> (greater-than)</td><td>></td></tr>
</table>

For Example:
Example 01:
If the value of Description attribute of a user account on Active Directory is
“My String & < > END". After reconcile the IBM Tivoli Identity Manager server fails this request with following error.

CTGIMD106E An error occurred while processing the request. Error: The content of elements must consist of well-formed character data or markup.

Here the character < (less than) is not transformed by ADK. This is because ADK found the substring & in the string and considered that the string is already transformed.

Example 02:
If the value of Description attribute of a user account on Active Directory is “My String & & > END". After reconcile the IBM Tivoli Identity Manager server fails this request with following error.

CTGIMD106E An error occurred while processing the request. Error: The entity name must immediately follow the '&' in the entity
</td></tr>
</table>

		reference. This is an error for "unknown entity section" because the "&" is assumed to begin an entity reference. Here the character & (ampersand) is not transformed by ADK. This is because ADK found the substring & in the string and considered that the string is already transformed. Example 03: If the value of Description attribute of a user account on Active Directory is “My String &amp & < END". After reconcile the IBM Tivoli Identity Manager server fails this request with following error. CTGIMD106E An error occurred while processing the request. Error: The reference to entity "lt" must end with the ';' delimiter. The first &amp doesn't end with a semicolon ';'. This is because ADK found the substring & in the string and considered that the string is already transformed. **Workaround:** 1. Modify the attribute value in such a way that either all the special characters in the value are replaced by their corresponding XML transformation or none are. Example 04: If the value of Description attribute of an user account on Active Directory is "My String & END" After reconcile the value displayed on IBM Tivoli Identity Manager account form is "My String & END". This will not cause any error or failure in the recon but the value displayed on Tivoli Identity Manager’s Account form is not the same as what is on Active Directory. This is because Tivoli Identity Manager does a reverse transformation to get the original character. In this case & will be replaced with & on Tivoli Identity Manager 2. Avoid the use of above special characters in attribute value.
		PMR 07331,035,724 - ITIM WinAD64 adapter - prob with WTS attrib When we perform the RECON operation, WinAD adapter returns 0x8000500d (ADS_PROPERTY_NOT_FOUND) error for all WTS attributes. This error is returned for those users which are directly created on Active Directory. This error will occur if you try to access attributes that aren't located in the so-called property cache. It could also be an operational attribute that isn't automatically built in the cache. Windows Active Directory adapter log will display following error message for all WTS attributes: **Error Message: "Failed with Error: 0x8000500d -**

		(null)" Microsoft has confirmed that this is a known issue in Windows Server 2008 Machine. For more information on this issue, Please visit the following Microsoft MSDN Web site: **http://support.microsoft.com/kb/947729** **Workaround:** Set any WTS attribute on Windows Server 2008 domain controller for those users which are giving the above error message. To set WTS attributes follow these steps: 1. Open **Active Directory Users and Computers**. 2. Find the user in the Users folder or in the **Organizational Unit** where the user is. 3. Right-click the user account, and then click **Properties**. 4. On the **Environment** tab or **Terminal Services Profile** tab, you will find the settings for WTS attributes value. 5. Doing a single modification on any one of these WTS attribute will resolve the issue. OR Customer can also follow the steps provided in the Microsoft MSDN web site as a workaround OR Running a modify request with WTS attributes from ITIM WinAD Service will also resolve the issue.
		Issue with WTS attributes during reconciliation operation Windows AD adapter gives following errors when managing WTS attributes and adapter registry key "WtsDisableSearch" set to FALSE. Error occurs when reconciliation is performed. **Could not reconcile WTS attribute erADWTSCallbckNumber. Failed with error 1317: The specified user does not exist. Could not reconcile WTS attribute erADWTSInheritInitialProg. Failed with error 1317: The specified user does not exist. Could not reconcile WTS attribute erADWTSRemoteHomeDir. Failed with error 1317: The specified user does not exist. Could not reconcile WTS attribute erADWTSCallbckSettings. Failed with error 1317: The specified user does not exist.** **Workaround:** no workaround.
		PMR 75002,442,000 Windows AD adapter gives following errors when trying to change value of existing mailboxstore attribute with Exchange 2003sp2. **Attribute erademailboxstore Condition code 5 (Error setting attribute erademailboxstore. ADSI Result code: 0x8000ffff - Catastrophic failure** **Error Moving the Mailbox. ADSI Result code: 0x8000ffff - Catastrophic failure** This condition may occur if "Administration User Account" and

		"Administration User Password" is given on AD service form on ITIM side and AD adapter is running either as service or in console mode, IMailboxStore::MoveMailbox () Method Fails The issue results from a restriction in Active Directory APIs due to security context when moving a user mailbox on a different exchange server. This is an expected behavior with the CDOEXM.DLL file. When you provide a credential by using the IAds::AdsOpenObject function, you cannot connect to a domain controller on a different computer by using the CDOEXM.DLL file. Instead, we recommend that you leave administrator username and password field blank on AD service form on ITIM side and run "Tivoli Active Directory Agent" windows service under domain administrator account by configuring windows service "Log on as:" property then mailbox gets moved successfully. Steps to configure "Tivoli Active Directory Agent" windows service: 1. Open the windows Services panel. Select "Tivoli Active Directory Agent" and click on "Properties". 2. Click on "Log On" tab. Select "This account" and enter domain administrator username and password. Click on "OK" to close the properties page.
		Leaving a space in the base point will cause problems The BasePoint (on Service form) and the Event Notification Context (in agentCfg) must be in proper DN format and should not contain any space between dc and ou\dc. For Example: If your basepoint value is <DC Name>/ou=testorg01,dc=testlab, dc=com //Here we have space between "dc=testlab," and "dc=com" Then you will have to specify correct baspoint value on service form in the following way: <DC Name>/ou=testorg01,dc=testlab,dc=com
		Special characters in the BasePoint attribute. The BasePoint that is specified on service form and for the Event Notification Context (in agentCfg) must be in proper DN format. Special characters like '**#**' '**+**' '**=**' '\' ';' '"' ',' '<' '>'in the base point value must be escaped with the escape character '\'. For Example: Example 1 If the basepoint to be specified is ADServer1/ou=#test<org>=01/end,dc=mydomain,dc=com The characters # < > = should be escaped using the \ character as ADServer1/ou=****#test****<org****>\=01/end,dc=mydomain,dc=com Example 2 If the base point to be specified is ou=inner;most,org",ou=outer+\org,dc=mydomain,dc=com The characters ; , " + and \ should be escaped using the \

		character as ou=inner\;most\,org\",ou=outer\+\\org,dc=mydomain,dc=com NOTE: Escaping of the '/' character is internally handled by adapter and should not be explicitly escaped on service form. It is recommended that the use of special characters should be avoided to get the best performance on various operations performed by adapter.
		Adapter based event notification not able to notify updates in attribute of type DNWithBinary to IBM Tivoli Identity Manager. You have extended adapter to manage attribute of type *DNWithBinary*. You have USER1 who's attribute of type *DNWithBinary*, for example, **otherWellknowObjects** contain DN of USER2. If you move or rename USER2 thereby changing DN, the change will be reflected in **otherWellknownObjects** attribute of USER1. The "object DN" part of value related to USER2 will be updated with new DN. The adapter based event notification is not able to identify this change ser1 **Example:** Consider we have two users USER1 and USER2. USER1's **otherWellknownObjects** attribute contains B:32:df447b5eaa5b11d28d5300c04f79ab81: CN=USER2,ou=myou,dc=mydomain,dc=com If you move USER2 to different container say myou2, the DN of USER2 changes to CN=USER2,ou=myou2,dc=mydomain,dc=com Active Directory updates the USER1's **otherWellknownObjects** with new DN of USER2 as B:32:df447b5eaa5b11d28d5300c04f79ab81: CN=USER2,ou=myou2,dc=mydomain,dc=com Even though USER1 is modified in such scenario it is not get notified by the adapter based event notification. The value of the extended attribute is not changed on IBM Tivoli Identity Manager. It will show old value i.e. B:32:df447b5eaa5b11d28d5300c04f79ab81: CN=USER2,ou=myou,dc=mydomain,dc=com **Workaround:** To update modified user i.e. USER1 on IBM Tivoli Identity Manager in such scenario we need to perform lookup for USER1 or full Recon.

		Adapter based event notification not able to notify updates to IBM Tivoli Identity Manager when all values of attribute are deleted directly on Active Directory Active Directory treats some attributes differently for example: *mail*, *info*, *otherWellknownObjects*, *msRTCSIP-UserPolicy* etc. For such attributes when all the values of the attribute are deleted from Active Directory, the attribute also get deleted from the user object. Since the attribute is deleted from Active Directory, The adapter based event notification does not identify such deleted attribute and so IBM Tivoli Identity Manager is not updated. **Workaround:** To clear all values from the IBM Tivoli Identity Manager perform full recon or Lookup operation

Installation and Configuration Notes

See the IBM Tivoli Identity Manager Adapter Installation Guide" for detailed instructions.

Running v4.6 and v5.0 Adapters on the Same Server

The Identity Manager version 5.0 adapters have enhanced capabilities that are not compatible with older version 4.6 adapters. It is highly recommended that all adapters hosted on an individual server are upgraded at the same time.

Adapters installed on the same server may share common components or run-time environments. The version 4.6 adapters may not be compatible with the version 5.0 component and may no longer operate as expected after installation of a version 5.0 adapter. On Windows servers all adapters must be upgraded simultaneously due to the sharing of DLLs. Check the adapter installation guide for additional information.

Corrections to Installation Guide

The following corrections to the Installation Guide apply to this release:

IMPORTANT NOTE:
The BasePoint (on Service form) and the Event Notification Context
(in agentCfg) must be in proper DN format. All special characters must be escaped.

Chapter 4. Installing the adapter -> Procedure

1. If you downloaded the installation software from Passport Advantage, perform the following steps:
   a. Create a temporary directory on the computer on which you want to install the software.
   b. Extract the contents of the compressed file into the temporary directory.
2. Start the installation program with the setup.exe file in the temporary directory.
3. Select the language and click OK to display the Introduction window.
4. On the Introduction window, click Next to view the Software License Agreement.
5. Do the following at the Software License Agreement window:
   - Review the license agreement and select Accept .
   - Click Next.
6. Select either Full installation or Update installation and click Next.
   Note: The adapter must already exist if you want to perform an update installation. If it does not exist, the installer generates the following message:
   Update not supported when there is no prior installation present on the system.

Cannot perform Update Installation. IBM Tivoli Windows Active Directory Adapter (32bit) does not exist on this system. Please select Full Installation.

7a. When you select Full installation option, specify where you want to install the adapter in the "Where Would You Like to Install?" field on Choose Install Folder. Do one of the following.
- Click Next to accept the default location.
- Click Choose and navigate to a different directory and click Next to view the pre-Installation Summary.

7b. When you select Update Installation option, then the adapter will display the path of the adapter installation which will be updated. Click OK to view the pre-Installation Summary.

8. Review the installation settings on the pre-Installation Summary window and do one of the following:
- Click Previous and return to a previous window to change any of these settings.
- Click Install when you are ready to begin the installation.

9. Click Done on the Install Complete window.

Chapter 12. Uninstalling the adapter -> Uninstalling the adapter from the target server

To remove the adapter, complete these steps:

1. Run the uninstaller. To run the uninstaller:
   a. Navigate to the adapter installation directory. For example, C:\Tivoli\agents\ADAgent_uninst
   b. Double click the uninstaller.exe file.
   Note: If you have installed the adapter using the -i silent option, then the uninstaller will be executed in silent mode.
2. Click Uninstall on Uninstall IBM Tivoli Windows Active Directory (32bit).
3. Check the status of uninstallation on the Uninstall Complete window.
4. Click Done.
5. Inspect the directory tree for the adapter directories, subdirectories, and files to verify that uninstall is complete.

Changing protocol configuration settings -> Table 5. Options for the DAML protocol menu -> Option 'k'

Modify Property 'READ_TIMEOUT':

Type the time out value for Tivoli Identity Manager and the adapter connection in seconds.

This applies to setups that have a firewall between Tivoli Identity Manager and the adapter. This firewall has a time out value that is less than the maximum connection age DAML property on Tivoli Identity Manager. When your transactions run longer than the firewall time out,
the firewall terminates the connection. The sudden termination of connections might leave the adapter with incorrect connection threads causing the adapter to crash.

When the adapter crashes randomly because of the specified setup, change the value for the READ_TIMEOUT. The value must be in seconds and less than the firewall's time out value.

Correction to Chapter 4: Troubleshooting the Active Directory Adapter

The following new section "Troubleshooting the Active Directory Adapter" expands the existing chapter 4 content.

Chapter 4. Troubleshooting the Active Directory Adapter errors- > Active Directory Adapter errors
Error message
Error: Could not retrieve WTS attribute inherit initial program. Error 87: The parameter is incorrect

Recommended action:
A cause of this error is because the target server specified in Base Point DN on service form does not run Windows Terminal Services.
Ensure that the target server specified for Base Point DN has Windows Terminal Services running.

Chapter 4. Troubleshooting the Active Directory Adapter errors- > Active Directory Adapter errors
(For IZ67106 - INTERMITTENT 0X80004002 ERRORS WHEN ATTEMPTING TO MANAGE/PROVISION MAILBOXES)
Error message
errorMessage="Unable to contact Exchange services. ADSI Result code: 0x80004002"

Recommended action:
The Exchange provider uses Collaboration Data Objects for Exchange Management (CDOEXM) for a user object. CDOEXM makes use of several static variables, since the lifetime of these variables last until process end. These static variables were being reallocated every time CDOEXM was loaded.
Since all CDOEXM work was done in the lifetime of the worker thread, CDOEXM was being loaded and unloaded repeatedly. Under certain conditions, CDOEXM is incorrectly marked as initialized, though CDOEXM is not fully initialized. Therefore, later attempts to use CDOEXM do not succeed.
You can use the new feature "Thread Pooling" of Windows Active Directory Adapter.

Additional information can be found under the "Configuration Notes->Enable/Disable "UseThreadPooling"" section.

Chapter 4. Troubleshooting the Active Directory Adapter errors- > Active Directory Adapter errors (for PMR 33834,999,760 - WinADAdapter failed to load exschema.txt at system bootup -> Table 8)
Correction to Recommended Action for error message "Error binding to schema container error code. Loading of extended schema attribute *attribute name* failed."

Error message	Recommended action
Error binding to schema container *error code*. Loading of extended schema attribute *attribute name* failed.	These errors occur when the Active Directory Adapter fails to extract the schema of the extended attributes. When the adapter service is started, the

	adapter reads exschema.txt and binds to the default domain i.e. domain in which adapter is running to check the syntax of the specified attribute. Since checking the syntax of extended attribute is one time process it is done at the startup. If adapter fails to bind to the default domain then it will not manage any of the extended attributes. Ensure that: 1. Ensure that the Active Directory is reachable from the workstation where the adapter is installed. 2. Verify that the extended attribute is correctly defined and added to the user class. 3. At least one domain controller is accessible before starting Active Directory adapter service. 4. The user account under which the adapter service is running has permission to read the Active Directory schema.

Configuration Notes

The following configuration notes apply to this release:

Note that the “supported configurations” diagrams have been moved into the Installation Guide.

Adding a Primary Group

To add a primary group to a user, the user must be a member of the group.

Delete Roaming Profile On Deprovision

If the delRoamingProfileOnDeprovision key is set to TRUE, the key enables user profile directory deletion when the user is de-provisioned. After successfully deleting the user from the Active Directory, the adapter deletes the user profile directory, subdirectories, and files. If this key is set to FALSE, or if the key does not exist, the adapter does not delete the user profile directory.

Deleting a Mailbox

To delete a Mailbox, delete the Alias attribute. Submit a reconciliation request to clear unnecessary values from the account form and to verify that the mailbox has been removed.

Proxy Address Configuration

A primary proxy address for each type must be added before additional proxy addresses of the same type can be added. The Win AD Agent requires that the Primary SMTP Address be set before setting the X.400 email address in proxy addresses field.

Directory NTFS and Share Access

The Agent returns the actual, effective permissions granted to a user and not the specific access assigned to the user account. For example, if the directory grants FULL permission to the Everyone group but only CHANGE permission to the user's account, a reconciliation request will return the account access permission as FULL. Therefore, it is necessary to properly define the policies local to the managed resource prior to using Tivoli Identity Manager to prevent these types of conflicts.

Expiration Date

Per Microsoft's documentation, the Active Directory Users and Computers MMC snap-in will display the account expiration date as one day earlier than the date contained in the accountExpires attribute. The Tivoli Identity Manager Server will display the value contained in the account expires attribute.

Password Properties

The password properties are specific to the account. However, these properties can be overridden by the security policies of the managed resource (Domain Controller Security Policies, Domain Security Policies, and Local Security Policies).

Setting Language Preference for Accounts

The Languages attribute (eradlanguage) is an Exchange attribute. If using a configuration without Exchange, setting this attribute will return a warning.

Log Message: Error More Data

NOTE: If a Reconciliation is run while the Active Directory server is under load, a logging message may appear in the WinAD Adapter log that says, "Error_More_Data." The Adapter is designed to retry the query three times before terminating the Reconciliation. Please see the Microsoft Knowledge base article below for more information.

When the IDirectorySearch::GetNextRow function returns S_ADS_NOMORE_ROWS, it may not have retrieved all the data from the server. In some cases, S_ADS_NOMORE_ROWS is returned by GetNextRow function when the server was unable to find an entry that matched the search criteria within a predefined two-minute time limit. This two-minute time limit is defined by means of an LDAP policy.

If the server exceeds the two-minute time limit, it returns an LDAP cookie in the response so that you can restart the search where it left off. Inefficient searches and heavily loaded systems can cause the server to exceed the time limit. When the server cannot find an efficient index to search, the server may have to apply the filter to every object in the directory, in which case it can run through many entries and not find a match within the two-minute time limit.

Therefore, when returning S_ADS_NOMORE_ROWS, ADSI also sets an extended error code, which can be queried using ADsGetLastError function. If ADsGetLastError returns ERROR_MORE_DATA, it means that the server has not completed the query and must call GetNextRow again.

The AD Agent code is structured as per the logic above and what Microsoft has advised. It attempts to get data from the paged result in max 3 attempts. If the AD Agent is running on AD server itself, Moving the AD Agent onto a different machine would take off some load from AD server.

In addition to this Microsoft has provided an article as how to configure the LDAP policy so as to customize the Active Directory searches.
http://support.microsoft.com/kb/315071/EN-US/.

Use SSL Configuration Option

The registry setting "useSSL" is to enable SSL communicating between AD Agent and Active Directory. If TRUE, agent communicates over SSL with Active Directory. If this key is not present or FALSE, agent does not use SSL.

By default this key is set to FALSE. No ITIM side changes are required to use this enhancement.

Following resource side changes are required to use this feature.

a. Active Directory must have enabled Public Key Infrastructure (PKI). For this Enterprise Certificate Authority should be installed on one of the domain controller machine in the domain. Setting up an enterprise certificate authority causes an Active Directory server to get a server certificate that can then be used to do SSL-based encryption.

b. Machine on which AD Agent is running should have certificate installed. The certificate is issued by CA as mentioned in point (a).

Use Default DC Configuration Option

The useDefaultDC registry setting is to provide failover capability to agent when host specified in base point is down. If agent is unable to connect to hostname specified in base-point and key is set to TRUE, agent will connect to the base-point without the host name. If it still fails then agent will report failure. By setting this key to TRUE also affects behavior of RAS server and Terminal server lookup.

Caution: When the adapter is deployed in a cross-domain scenario, the useDefaultDC option should always be set to FALSE to avoid provisioning to an unintended domain. For example, if the adapter is installed in domain A but provisioning to domain B, and the host in domain B is down, the adapter will detect the default domain as domain A.

By default this key is set to FALSE.

The behavior of agent will be as follows:

A. useDefaultDC = FALSE

   i. If hostname (target server name) is specified in the base point and ForceRASServerLookup and ForceTerminalServerLookup registry keys are set as FALSE, then agent uses the given hostname as RAS server and Terminal server.

   ii. If hostname (target server name) is specified in the base point and ForceRASServerLookup and ForceTerminalServerLookup registry keys are set as TRUE, then agent will determines the RAS and Terminal server name.

B. useDefaultDC = TRUE and host is down.

   i. Agent will determines RAS and Terminal server irrespective of values set for ForceRASServerLookup and ForceTerminalServerLookup.

Use new Win2003 ADSI API for managing WTS attributes

Agent will use WTS ADSI API's or old style WTS API's to set or retrieve WTS attributes. Agent will try to use WTS ADSI API's, if it fails to get interface or attribute is not supported then agent will use old style WTS API's.

If agent is running on Windows 2003 then agent will use WTS ADSI API's. On Windows 2000 agent will use old style WTS API's.

From log it can be found out which WTS API's agent is using. Some of the attributes are not supported by WTS ADSI API's; for that agent will use old style WTS API's on Windows 2000 and Windows 2003.

If debug logging is enabled, then agent will show lines like:

- *Start using extended interface for WTS Attributes for getting WTS attribute.*
- *Start using extended interface for WTS Attributes for setting WTS attribute.*

- *End using extended interface for WTS Attributes for setting WTS attribute.*
- *End using extended interface for WTS Attributes for getting WTS attribute.*

This means agent is using WTS ADSI API's.

If log is showing lines like:

- *Using old style API for WTS Attributes for getting WTS attribute*
- *Using old style API for WTS Attributes for setting WTS attribute*

This means agent is using old style WTS API's.

Win AD Agent handle Add and Delete operations for erGroup attribute

WinAD Adapter by default honors only replace operation from Tivoli Identity Manager. WinAD Adapter now supports Add and Delete operation for erGroup attribute for Modify request.

On Tivoli Identity Manager profile changes are required in ADprofile to send group operation as add, delete for modify request.
See the steps below

- Locate the following line in resource.def file under <Operation Name="modify"> tag.

<Parameter Name="erGroup" Source="account" ReplaceMultiValue="true" />

- Replace the above line with following line.

<Parameter Name="erGroup" Source="account" />

- Reinstall the profile. Please see Windows Active Directory Adapter Install guide for further references.

Running Mixed Versions of Active Directory and Exchange

The agent is certified to run in Windows 2003 Active Directory and Exchange 2000 Exchange server environment. Windows 2003 Active Directory and Exchange 2000 on same server is not a valid configuration. These must be installed on separate machines.

The following limitations apply to this mixed mode environment:

1) A domain controller must be specified as a part of base point to create mailbox on the Member server.
2) Mailbox move is not supported in a mixed mode environment. Move mailbox operation will fail when an attempt is made to move a mailbox from or to a mailstore on the member server.
    a. Move mailbox from a Mailstore on Member server to a Mailstore on Primary DC **fails.**
    b. Move mailbox from a Mailstore on Member server to another Mailstore on Member server **fails.**
    c. Move mailbox from a Mailstore on Primary DC to another Mailstore on Primary DC **succeeds.**

Using DN or GUID for the erGroup Attribute

Windows Active Directory Adapter has been enhanced to support Group DN and Group GUID values for erGroup attribute. With this enhancement, multiple groups with the same name that are present in different organizational units on active directory can be processed by the adapter.

Beginning with this version of the adapter, the new registry key "UseGroup" is introduced. The default value for this key is CN. The registry key value can be set to CN or DN or GUID as per the requirement.

Configuration required for using this feature:
Perform the following steps:

A] Set the UseGroup registry key to CN or DN or GUID using agentCfg.

B] Change the profile files “erADAccount.xml” and “resource.def”

Find the table below describing what change need to be done when UseGroup registry value is set to CN or DN or GUID

UseGroup value	**Change required in**
CN It is default value	**erADAccount.xml** <formElement name="data.ergroup" label="$ergroup"> <searchFilter multiple="true" type="select"> <filter>(objectclass=eradgroup)</filter> <base>contextual</base> **<attribute>eradgroupcn</attribute>** **<sourceAttribute>eradgroupcn</sourceAttribute>** <size></size> <objectClass></objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement> **Resource.def** <ServiceGroups> <GroupDefinition ProfileName="ADGroupProfile" ClassName = "erADGroup" RdnAttribute = "erADGroupGUID" AccountAttribute = "erGroup"> <AttributeMap> **<Attribute Name="erGroupId" Value="erADGroupCN" />** **<Attribute Name="erGroupName" Value="erADGroupCN" />** <Attribute Name="erGroupDescription" Value="description" /> </AttributeMap> </GroupDefinition> </ServiceGroups>

DN **erADAccount.xml**

```
<formElement name="data.ergroup" label="$ergroup">
<searchFilter multiple="true" type="select">
<filter>(objectclass&#61;eradgroup)</filter>
<base>contextual</base>
<attribute>eradgroupdn</attribute>
<sourceAttribute>eradgroupdn</sourceAttribute>
<size></size>
<objectClass></objectClass>
<showQueryUI>false</showQueryUI>
<paginateResults>false</paginateResults>
</searchFilter>
</formElement>
```

Resource.def

```
<ServiceGroups>
<GroupDefinition ProfileName="ADGroupProfile"
ClassName = "erADGroup"
RdnAttribute = "erADGroupGUID"
AccountAttribute = "erGroup">
<AttributeMap>
<Attribute Name="erGroupId" Value="erADGroupDN" />
<Attribute Name="erGroupName" Value="erADGroupDN" />
<Attribute Name = "erGroupDescription" Value="description" />
</AttributeMap>
</GroupDefinition>
 </ServiceGroups>
```

GUID **erADAccount.xml**

```
<formElement name="data.ergroup" label="$ergroup">
<searchFilter multiple="true" type="select">
<filter>(objectclass&#61;eradgroup)</filter>
<base>contextual</base>
<attribute>eradgroupdn</attribute>
<sourceAttribute>eradgroupguid</sourceAttribute>
<size></size>
<objectClass></objectClass>
<showQueryUI>false</showQueryUI>
<paginateResults>false</paginateResults>
</searchFilter>
</formElement>
```

Resource.def

```
<ServiceGroups>
<GroupDefinition ProfileName="ADGroupProfile"
ClassName = "erADGroup"
RdnAttribute = "erADGroupGUID"
AccountAttribute = "erGroup">
<AttributeMap>
<Attribute Name="erGroupId" Value="erADGroupGUID" />
<Attribute Name="erGroupName" Value="erADGroupDN" />
<Attribute Name = "erGroupDescription" Value="description" />
</AttributeMap>
</GroupDefinition>
</ServiceGroups>
```

NOTE:

1] The value of the “attribute” tag of the formElement “data.ergroup” in erADAccount.xml should match with the value of “erGroupName” in resource.def.

2] The value of the “sourceAttribute” tag of the formElement “data.ergroup” in erADAccount.xml should match with the value of “erGroupId” in resource.def

3] Keep the value of the “sourceAttribute” tag of the formElement “data.eradprimarygroup” unchanged to “eradprimarygrptkn”. This value has to remain unchanged for CN, DN or GUID support.

C] Build the ADprofile.jar and import the new profile into TIM

D] Run a full reconciliation.

NOTE: If an event notification is enabled, delete the event notification data base using the agentCfg (Refer Install guide for the deleting the event notification database) and then run a full reconciliation. This will make sure that new database get created with correct values for attribute “erGroup”

Installing the Windows Active Directory Adapter by using silent mode

You can install and uninstall the IBM Tivoli Windows Active Directory Adapter (32bit) by using silent mode. Silent installation suppresses the Wizard and the Launcher User Interfaces (UIs) that do not display any information or require interaction.
You can use the -i silent option to install or uninstall the adapter in silent mode.

Note:

i. The adapter installer also installs run time libraries from Microsoft. The user interface of the installer for these run time libraries is also suppressed during silent installation of the adapter. The installer for these run time libraries creates a log file “vcredist_x86.log” under the temp directory of the user home directory (%temp%). For example, “C:\Documents and Settings\Administrator\Local Settings\Temp\vcredist_x86.log”. It is recommended that you check this file for any errors.
ii. If you install adapter in silent mode, the uninstaller runs in silent mode irrespective of whether you are using -i silent option or not.

Installing the adapter by using the silent mode

Installing the adapter with default options

Run the following command from command line to install the Windows Active Directory Adapter by using the -i silent option:

setup.exe -i silent -DLICENSE_ACCEPTED=TRUE

The adapter is installed with the following default values.

Installation Directory	%SYSTEM_DRIVE_ROOT%\Tivoli\agents\ADAgent
Installation Type	Full Installation

Installing the adapter with command line parameters

You can specify installation options from the command line when you install the adapter by using the silent option. The following table lists the parameters used by the installer. Note:

- The -D option is followed by a variable and a value pair without any space after -D.
- You must wrap arguments with quotation marks when the arguments contain spaces.

Parameter Name	Description	Default Value
DLICENSE_ACCEPTED	The installer uses this parameter to get the license acceptance state. When TRUE is supplied as the value for this parameter, it indicates that you accept the terms in the license agreement of the adapter. If the value of this parameter is FALSE or if this parameter is missing then the installation will not continue. Note: - The parameter is must when you run the installer in silent mode.	FALSE
DUSER_INSTALL_DIR	The value of this parameter overrides the default installation directory path. For example, DUSER_INSTALL_DIR="D:\Tivoli\MyFolder" Note: - The installation path must be wrapped in quotation marks.	%SYSTEM_DRIVE_ROOT%\Tivoli\agents\ADAgent
DUSER_INPUT_INSTALL_TYPE_1	When the value of this parameter is \"Full Installation\" the installer performs full installation of the adapter. For example, DUSER_INPUT_INSTALL_TYPE_1=\"Full Installation\"	\"Full Installation\"
DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1	This parameter is associated with DUSER_INPUT_INSTALL_TYPE_1. When the value of this parameter is 1 the installer performs full installation of the adapter. You can either use DUSER_INPUT_INSTALL_TYPE_1 or DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1 or both to perform full installation.	1
DUSER_INPUT_INSTALL_TYPE_2	When the value of this parameter is \"Update Installation\" the installer performs update installation of the adapter. For example, DUSER_INPUT_INSTALL_TYPE_2=\"Update Installation\" Note:- When \"Update Installation\" is specified as the value for this parameter, the parameter DUSER_INPUT_INSTALL_TYPE_1 should be set to blank or should not be specified at all and you must explicitly override the default value of DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1 to 0.	No value, blank
DUSER_INPUT_INSTALL_TYPE_	This parameter is associated with	0

BOOLEAN_2	DUSER_INPUT_INSTALL_TYPE_2. When the value of this parameter is 1 the installer performs update installation of the adapter. You can use either DUSER_INPUT_INSTALL_TYPE_2 or DUSER_INPUT_INSTALL_TYPE_BOOLEAN_2 or both to perform update installation. Note:- When the value 1 is specified as the value for this parameter, the parameter DUSER_INPUT_INSTALL_TYPE_1 should be set to blank or should not be specified at all and you must explicitly override the default value of DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1 to 0.	

You can use any of the following commands to perform a full installation of the adapter in silent mode.

1. setup.exe -i silent -DLICENSE_ACCEPTED=TRUE

   This will install the adapter using Full Installation option and will use the default installation directory %SYSTEM_DRIVE_ROOT%\Tivoli\agents\ADAgent, as stated above.

2. setup.exe -i silent -DLICENSE_ACCEPTED=TRUE -
   DUSER_INPUT_INSTALL_TYPE_1=\"Full Installation\" -
   DUSER_INPUT_INSTALL_TYPE_2= -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1=1 -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_2=0

   This will install the adapter using Full Installation option and will use the default installation directory %SYSTEM_DRIVE_ROOT%\Tivoli\agents\ADAgent.

   To select a different installation directory add the -DUSER_INSTALL_DIR parameter as:

   setup.exe -i silent -DLICENSE_ACCEPTED=TRUE -
   DUSER_INSTALL_DIR="C:\Tivoli\MyFolder" -
   DUSER_INPUT_INSTALL_TYPE_1=\"Full Installation\" -
   DUSER_INPUT_INSTALL_TYPE_2= -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1=1 -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_2=0

3. setup.exe -i silent -DLICENSE_ACCEPTED=TRUE -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1=1 -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_2=0

   This will install the adapter using Full Installation option and will use the default installation directory %SYSTEM_DRIVE_ROOT%\Tivoli\agents\ADAgent.

   To select a different installation directory add the -DUSER_INSTALL_DIR parameter as:

   setup.exe -i silent -DLICENSE_ACCEPTED=TRUE -
   DUSER_INSTALL_DIR="C:\Tivoli\MyFolder" -

DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1=1 -
DUSER_INPUT_INSTALL_TYPE_BOOLEAN_2=0

Installing the adapter in silent mode by using the response file

Generating the response file

You can use response file to provide inputs during silent installation. Response file can be generated by running the following command.

setup.exe -r "Full path of response file"

For example:
setup.exe -r "C:\Temp\ WinADInstallParameters.txt"

This runs the installer in interactive mode and installs the adapter.
After the installation completes the file specified as "Full path of response file" will be created containing the required parameters.

Note:

- If you are running this command to only generate the response file, you must uninstall the adapter by using the uninstaller.

Creating the response file manually

You can also manually create the response file and add the required parameters to the file.

Create a text file, for example C:\WinADInstallParameters.txt, with the following content:

```
#Has the license been accepted
#-----------------------------
LICENSE_ACCEPTED=TRUE

#Select Install Type
#-------------------
USER_INPUT_INSTALL_TYPE=\"Full Installation\",\"\"
USER_INPUT_INSTALL_TYPE_1=Full Installation
USER_INPUT_INSTALL_TYPE_2=
USER_INPUT_INSTALL_TYPE_BOOLEAN_1=1
USER_INPUT_INSTALL_TYPE_BOOLEAN_2=0

#Choose Install Folder
#--------------------
USER_INSTALL_DIR=C:\\Tivoli\\agents\\ADAgent
```

After you have the response file you can use it to provide parameters to the installer for silent installation as:

setup.exe -i silent -f "Full path of response file"
For example,
setup.exe -i silent -f "C:\WinADInstallParameters.txt"

Updating the Active Directory Adapter or the ADK

Note:

- If your existing adapter version is earlier than 5.0.x, you must uninstall the older version of the adapter before you can install the 5.0 adapter. You cannot migrate from a version earlier than 5.0 to 5.0 because the encryption used in the 5.0 release is not compatible with previous versions of the ADK. Any previously encrypted values cannot be read by the 5.0 adapter.
- This version of 5.0 adapter uses a newer version of the installer. It is recommended that you uninstall the existing adapter, if the existing adapter build number is 5.0.1020 or older.

You can either update the Windows Active Directory adapter or the Adapter Development Kit (ADK). The ADK is the base component of the adapter. While all adapters have the same ADK, the remaining adapter functionality is specific to the managed resource.

If only a code fix has been made to the ADK, instead of upgrading the entire adapter, you can upgrade just the ADK to the newer version. See "Updating the ADK" in Installation Guide.

Updating the Windows Active Directory adapter

Note: Update adapter is not supported if you are updating from adapter build number 5.0.1020 or older.

For adapter versions 5.0 and higher, use the adapter upgrade option:

- If you want to keep the adapter configuration (registry keys and certificates) unchanged.
- If the installed adapter is FIPS enabled. The Update Installation option keeps FIPS configurations such as the CA certificates, fipsdata.txt the (key generated by running fipsenable.exe) and the registry keys encrypted with fipsdata.txt unchanged.

If update installation option is selected, the installer detects the path of the existing installed adapter. If no prior installation of the adapter is found on the system, the installer will display an error message. The installer replaces the binaries and the DLLs of the adapter and the ADK. The installer does not prompt for any configuration information during an update installation.

Note: Adapter related registry keys are not modified. The update installation does not create a new service for the adapter.

During an update, in order to maintain all of your current configuration settings, as well as the certificate and private key, do not uninstall the old version of the adapter before installing the new version. For more information on how to install the adapter, see "Installing the Windows Active Directory adapter" in Installation Guide.

In order to update an existing adapter, complete the following steps:

7. If you downloaded the installation software from Passport Advantage, perform the following steps:
   - Create a temporary directory on the computer on which you want to install the software.
   - Extract the contents of the compressed file into the temporary directory.
8. Start the installation program with the setup.exe file in the temporary directory.
9. Select the language and click OK to display the Introduction window.
10. On the Introduction window, click Next to view the Software License Agreement.
11. Do the following at the Software License Agreement window:
    - Review the license agreement and select Accept .
    - Click Next.
12. Select Update installation option and click Next
    Note:
    The adapter must already exist if you want to perform an update installation. If it does not exist, the installer generates the following message:

Update not supported when the adapter is not previously installed.
Cannot perform Update Installation. IBM Tivoli Windows Active Adapter (32bit) does not exist on this system. Please select Full Installation.

13. The adapter will display the path of the adapter installation which will be updated. Click OK to view the pre-Installation Summary.
14. Review the installation settings on the pre-Installation Summary window and click on Install.
15. Click Done on the Install Complete window.

Updating Windows Active Directory Adapter by using silent mode

You can use the -i silent option to update the adapter in silent mode.
Please refer to the table under "Installing the Windows Active Directory Adapter by using silent mode" for detail information about installer command line parameter for silent installation.
Note:

- If you install adapter in silent mode, the uninstaller runs in silent mode irrespective of whether you are using -i silent option or not.
- When performing Update Installation in silent mode the -DUSER_INSTALL_DIR parameter must never be used
- When updating the adapter using silent installation, the parameter DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1=0 must be used to overwrite the default value of this parameter.
- The adapter installer also installs run time libraries from Microsoft. The user interface of the installer for these run time libraries is also suppressed during silent installation of the adapter. The installer for these run time libraries creates a log file “vcredist_x86.log” under the temp directory of the user home directory (%temp%). For example, “C:\Documents and Settings\Administrator\Local Settings\Temp\vcredist_x86.log”. It is recommended that you check this file for any errors

Updating the adapter with command line parameters

You can use any of the following commands to perform update installation of the adapter in silent mode.

1. setup.exe -i silent -DLICENSE_ACCEPTED=TRUE -
   DUSER_INPUT_INSTALL_TYPE_1= -
   DUSER_INPUT_INSTALL_TYPE_2=\"Update Installation\" -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1=0 -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_2=1

2. setup.exe -i silent -DLICENSE_ACCEPTED=TRUE -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1=0 -
   DUSER_INPUT_INSTALL_TYPE_BOOLEAN_2=1

Note:-
The installer itself detects if the adapter is already installed on the system on which this command is executed. For this the installer refers to the adapter registry keys.
The installer proceeds with updating the adapter only if it successfully detects a prior installation of the adapter on the system.
If no prior installation is found on the system, the installation is aborted and a log file IBM_Tivoli_Windows_Active_Directory_Adapter_(32_Bit)_InstallLog.log is generated with the following information on the Desktop:

Action Notes:
Cannot perform Update Installation, IBM Tivoli Windows Active Directory Adapter (32 Bit) does not exist on this system. Select Full Installation to install IBM Tivoli Windows Active Directory Adapter (32 Bit) on this system. Refer to installation guide on how to install the adapter in silent mode and with full installation.

Cannot perform Update Installation, IBM Tivoli Windows Active Directory Adapter (32 Bit) does not exist on this system.
Status: FATAL ERROR
Additional Notes: FATAL ERROR - UPDATE INSTALLATION NOT SUPPORTED

Updating the adapter in silent mode by using the response file

Generating the response file

You can use response file to provide inputs during silent installation. Response file can be generated by running the following command.

setup.exe -r "Full path of response file"

For example:
setup.exe -r "C:\WinADInstallParameters.txt"

This runs the installer in interactive mode and installs the adapter.
After the installation completes the file specified as "Full path of response file" will be created containing the required parameters.

Note: If you are running this command to only generate the response file, you must uninstall the adapter by using the uninstaller.

Creating the response file manually

You can also manually create the response file and add the required parameters to the file.

Create a text file, for example WinADInstallParameters.txt, with the following content:

```
#Has the license been accepted
#----------------------------
LICENSE_ACCEPTED=TRUE

#Select Install Type
#-------------------
USER_INPUT_INSTALL_TYPE=\"\",\"Update Installation\"
USER_INPUT_INSTALL_TYPE_1=
USER_INPUT_INSTALL_TYPE_2=Update Installation
USER_INPUT_INSTALL_TYPE_BOOLEAN_1=0
USER_INPUT_INSTALL_TYPE_BOOLEAN_2=1
```

After you create the response file you can use it to provide parameters to the installer for updating the adapter using silent installation as:

setup.exe -i silent -f "Full path of response file"
For example,
setup.exe -i silent -f "C:\WinADInstallParameters.txt"

Note: Restart the workstation after you install or uninstall the adapter.

When the installation completes with updating the adapter, new install log file will be created replacing the old file in the installation directory.

The adapter installer allows an update installation of the adapter, for adapter's versions 5.1 or later.

Uninstalling the adapter using silent mode

Run the following command from command line to uninstall IBM Tivoli Windows Active Directory Adapter (32bit) by using the -i silent option.

uninstaller.exe -i silent

Specify the full path when you are not running the command from _uninst directory in the installation directory of the adapter.

For example,
"C:\Tivoli\agents\ADAgent_uninst\uninstaller.exe" -i silent

Note:
- Restart the workstation after you install or uninstall the adapter.
- Silent un-installation may not completely clean the installation directory. There may be some files or folder which is not removed. Please check the installation folder and remove files and folder which are not required after the un-installation completes.

Behavior of 'mail' attribute

In Active Directory each User/Group object has a 'mail' attribute to store single e-mail address. With Exchange 2003/2007/2010 this property points to the primary SMTP address of that object. When the object is first mail- or mailbox-enabled, the "mail" attribute is set to the primary SMTP proxy address. The primary SMTP address itself is stored in the proxy Addresses field as part of the e-mail address list. If the value of primary SMTP proxy address is modified, then the "mail" attribute (E-mail address) is replaced with the new value of primary SMTP proxy address.
On Exchange 2003, If the value of “mail" attribute (E-mail address) is modified, the primary SMTP proxy is replaced with the new value of "mail" attribute (E-mail address). On Exchange 2007 or 2010 changing the value of “mail" attribute (E-mail address) does not have any effect on primary SMTP proxy address.
Proxy addresses are generated by Recipient Update Service. The value of 'mail' attribute corresponds to the primary SMTP proxy address. If a mail or mailbox enabled account is enabled then the old value of mail attribute gets cleared and is set with the new value of primary SMTP address. On Exchange 2003 if a mail or mailbox enabled account is disabled then value of mail attribute gets cleared. On Exchange 2007 or 2010 this value is not cleared if mail or mailbox account is disabled.

NOTE: It is recommended to have same value of primary SMTP proxy address and mail attribute in Active Directory to avoid unexpected behavior.

Please refer the section describing the 'mail' attribute in following MS links

http://support.microsoft.com/kb/275636

New Adapter Features

The following features have been added to the adapter in this release.

Adapter Version 5.0.3 Features

Password Sync Recursion Control

The recursion control for password synchronization has moved from the adapter to the Identity Manager server. This change allows high-availability configurations to be setup when using multiple WinAD adapters per domain.

Note : IBM Tivoli Identity Manager 5.0 Fix Pack 2 provides server side recursion control.

Before upgrading to Windows Active Directory Adapter build 5.0.1015, the registry key 'PasswordChanges' should be removed. Use the following steps to remove the key.

1. Run regedit or regedt32 utility.
2. Go to key HKEY_LOCAL_MACHINE\SOFTWARE\Access360\ADAgent\Specific, to find the registry key 'PasswordChanges'.
3. Right click on the 'PasswordChanges' key and select Delete.

Dial-In Options

A new attribute "erADExDialin" is added to support this enhancement. A new String attribute "erADExDialin" is added in erADAccount object class.

Attribute "erADExDialin" is added to the Active Directory accounts form (erADAccount.xml). The account form now has a combo box instead of a checkbox representing one the three values for dial-in.

Value displayed on account form	Value sent and stored in TIM LDAP
Allow Access	TRUE
Deny Access	FALSE
Control access through Remote Access Policy	NONE

Attribute "erADAllowDailin" is deprecated and will not be processed by AD Agent
Attribute "erADAllowDailin" is removed from the accounts from.

Note:

1) If you have any business logic around attribute "erADAllowDialin" (e.g. a Provisioning Policy Parameter) then you must modify it to use "erADExDialin".

2) In mixed-mode domain functional level third dial-in option "Control access through Remote Access Policy" is not supported.

If an attempt is made to set dial-in as "Control access through Remote Access Policy" agent will report attribute level failure to TIM, but on the resource dial-in will be set to "Deny Access". If previously dial-in was set to "Allow Access" after the above failed request dial-in will be set as "Deny Access".

lastLogonTimeStamp attribute

This version of Windows Active Directory Adapter supports lastLogonTimeStamp attribute of Active Directory. Attribute lastLogonTimeStamp is available on Windows 2003 domain functional level and is replicated. The default replication interval is 14 days.

To support this enhancement profile of Windows Active Directory Adapter is extended. A new Date attribute erADLastLogonTimeStamp (OID: 1.3.6.1.4.1.6054.3.125.2.146) is defined and added in erADAccount class. A new label (eradlastlogontimestamp=Last Login Time Stamp) is added to CustomLabels.properties file.

Note: The attribute erADLastLogonTimeStamp is not visible on account form. To bring it on account form, form customization is required.

Adapter Version 5.0.4 Features

Support for DL email Attribute

This version of Windows Active Directory Adapter supports “mail” attribute of Active Directory Groups. A new attribute erADGroupDlEmail (OID: 1.3.6.1.4.1.6054.3.125.2.147) is defined and added in erADGroup class. A new label (eradgroupdlemail=Distribution List e-mail) is added to CustomLabels.properties file. The attribute stores email address of the distribution groups.

If you want to distinguish between distribution list groups from security groups, then you can use this attribute which gets retrieved in reconciliation operation.

Single Transaction for Modifying Mail-user to Mailbox User

This version of Windows Active Directory Adapter provides an enhancement to modify a mail account from its existing mail status (e.g. Mailuser) to other mail status (i.e. Mailbox user) in one single ITIM Modify operation. In the earlier versions of WinAD adapter, to modify the mail status, one has to delete the existing mail status in one operation and create the new mail status in another operation.

- To modify a Mailuser account to a Mailbox account -- Clear the target address and specify mailbox store. You may modify the Alias if required. The mailuser status is removed and a mailbox is created for the account.
- To modify a Mailbox account to a Mailuser account -- Clear the mailbox store and specify target address. You may modify the Alias if required. The mailbox is deleted from the exchange and the account is mail-enabled.

NOTE: If the Alias is not modified during the above operations, then the Alias value is retained in the new mail status. To delete an existing mailbox (mailuser or mailbox) of mail account, delete the Alias attribute. (this behavior is same as previous versions of Adapter)

All other exchange related operations will work same like in earlier versions of Windows active directory adapter.

Support for Windows 2008

When you use Windows 2008 as a installation platform, and require to run adapter in SSL mode, then follow below steps. Otherwise, the certificate install will not be complete and will not enable SSL correctly.

1. Disable UAC security (User Account Control).
2. Install the required Certificate.
3. If required, enable UAC security.

For more information visit the link to enable/disable the UAC.

http://en.wikipedia.org/wiki/User_Account_Control

Adapter Version 5.0.5 Features

Bypassing Recon of Mailbox Security Attributes

Some customers experience AD Adapter recon crashes when parsing mailbox security descriptor data. A new registry key 'ReconMailboxPermissions' has been introduced to provide a workaround for this issue. This registry key is set to TRUE by default and will reconcile mailbox security permissions attributes. Recon and parsing of mailbox permissions attributes can be bypassed by setting registry key 'ReconMailboxPermissions' to FALSE. This will also improve reconciliation performance.

Adapter Version 5.0.6 Features

Failover of Target Systems – Multiple Servers in Basepoint

MR0212084734 – Enhancement to handle failover of target systems by supporting multiple target servers in basepoint.

This version of Windows Active Directory Adapter supports more than one target servers in the base point.

The earlier versions of WinAD adapter supported only one target server for the basepoint. If the specified target server is down and useDefaultDC is set to FALSE, adapter used to fail the request.

With this enhancement more than one target servers can be specified for basepoint on Tivoli Identity Manger's Active Directory Service Form and/or in the Windows Active Directory Adapter's registry. Each target server must be separated by a pipe '|'.

Usage Example,

Base Point DN on service form with more than one target server:
DC01|DC02|DC03/OU=TestOU,DC=MyDomain,DC=com

Base Point DN on service form with only one target server:
DC01/OU=TestOU,DC=MyDomain,DC=com

Base Point DN on service form with no target server:
OU=TestOU,DC=MyDomain,DC=com

Also, target servers can be specified in Adapter's registry as well:

a) If BasePoint specified on service form is “OU=TestOU,DC=MyDomain,DC=com”, you can specify the list of target server(s) in adapter registry using agentCfg.exe as follows:

1. Create WinAD adapter registry with name OU=TestOU,DC=MyDomain,DC=com
2. Specify the value for the above key as DC01|DC02|DC03

Registry key	Value
OU=TestOU,DC=MyDomain,DC=com	DC01|DC02|DC03

b) If BasePoint specified on service form is “DC01|DC02|DC03/OU=TestOU,DC=MyDomain,DC=com”, you can specify the list of target server(s) in adapter registry using agentCfg.exe as follows:

3. Create WinAD adapter registry with name OU=TestOU,DC=MyDomain,DC=com
4. Specify the value for the above key as DC01|DC02|DC03

Registry key	Value
OU=TestOU,DC=MyDomain,DC=com	DC01|DC02|DC03

Adapter iterates through all the target servers specified in base point on service form, and then through the target servers specified in registry key. The first available target server is used by the adapter.
You can also wish to specify the base point without the target server(s) on service form, and use the registry key to specify the target servers. Adapter uses the base point specified on the service form to find a key with this base point value in the registry, to get the target servers specified as the value for this registry key.

Note:

1. The maximum number of characters that the Base Point DN attribute on service form can hold is 240 characters.
2. Service form and registry can specify their own set of target servers, the target servers specified on service form is given high priority.
   Example,
   Base Point on service form
   DC01|DC02|DC03/OU=TestOU,DC=MyDomain,DC=com
   Registry

Registry key	Value
OU=TestOU,DC=MyDomain,DC=com	DC04|DC05|DC06

3. If no base point is specified on service form, the registry will not be referred.
4. It is recommended that you specify target server using adapter’s registry as it is cached to improve performance as against specifying on service form. The target server list on service form is not cached and is parsed in each request to find all target servers.
5. Use agentCfg.exe to create and modify adapter registry keys, please restart the adapter service after adding or modifying registry keys. If the basepoint or target server contains Unicode characters then use regedit to create registry keys under HKEY_LOCAL_MACHINE\ SOFTWARE\Access360\ADAgent\Specific

Proxy Address Attribute Change/Delete

MR032609161- WinAD Enhancement to handle erADEProxyAddresses attribute in add/delete as well as in replace format.

This version of Windows Active Directory Adapter is enhanced to support erADEProxyAddresses attribute values in an ADD\DELETE format.

Using the current profile, IBM Tivoli Identity Manager will send the erADEProxyAddresses attribute in the modify request with operation type as replace. In order to use this enhancement it is required that the adapter receives the erADEProxyAddresses attribute in the modify request in an add/delete format. The following changes are required in ADprofile.jar so that IBM Tivoli Identity Manager handle the erADEProxyAddresses attribute in add/delete format.

The profile JAR file, ADprofile.jar, is included in the Active Directory Adapter compressed file that you downloaded from the IBM Web site.

Complete the following steps to modify the ADprofile.jar file:

1. Copy the ADprofile.jar file to a temporary directory, for example C:\Temp folder.
2. Extract the contents of ADprofile.jar file into the temporary directory by running the following command:
   cd C:\Temp
   jar -xvf ADprofile.jar
   The jar command will create the "C:\Temp\ADprofile" directory which will have all the profile files.
3. For Editing ADprofile complete the remaining steps below

   I. For editing WinAD Adapter profile for IBM Tivoli Identity Manager version 4.6
      a) Open xforms.xml file found in the extracted ADprofile folder in text pad.
      b) Add the following entry between the tags <EnRoleTransformations> and </EnRoleTransformations>
         <EnRoleAttribute Name="erADEProxyAddresses"
         RemoteName="erADEProxyAddresses" ConvertReplaceToAddDelete="true" />
      c) Save the xforms.xml file

   II. For editing WinAD Adapter profile for IBM Tivoli Identity Manager version 5.0
      a) Open resource.def file found in the extracted ADprofile folder in text pad.
      b) Search for the entry <Parameter Name="erADEProxyAddresses" Source="account" ReplaceMultiValue="true" /> in resource.def file.
      c) Delete all the occurrences of the above entry form resource.def file.
      d) Save the resource.def file.

4. Use following commands to create new jar file
   cd C:\Temp
   jar -cvf ADprofile.jar ADprofile
5. Import the new ADprofile.jar file into IBM Tivoli Identity Manager Application server. For more information on importing the file, see section “Importing the adapter profile into the Tivoli Identity Manager server" in WinAD Adapter Installation and Configuration guide.
6. After importing the profile, stop and start the Tivoli Identity Manager server to reflect the changes.

Provisioning to Child Domain - UPN Checking

IZ38367 - Adapter provisioning to child domain UPN issue.

This version of Windows Active Directory Adapter provides an additional check for uniqueness of the User Principal Name (UPN) within a forest, while creating a user account.

The adapter now performs search on User Principal Name (which is either generated by the adapter or supplied in the user add request) in the forest to ensure that no two users have same User Principal Name. For this adapter performs a search first in the current managed domain and then in the forest.
Note:
The Current version of adapter does not perform User Principal Name (UPN) check for modify operation.

A new registry key “UPNSearchEnabled” is introduced to use this enhancement. This registry key is set to TRUE by default.
When registry key UPNSearchEnabled is set to FALSE, adapter will not perform a search on User Principal Name for uniqueness and will create the user account with the supplied or generated value of User Principal Name.
When registry key UPNSearchEnabled is set to TRUE, adapter works as follows:

Case 1: User Principal Name is supplied in the user add request.
Adapter will use the value of User Principal Name and will perform a search. If an account exists in the forest with the same User Principal Name, adapter will fail the user add request.

Case 2: User Principal Name is not supplied in the user add request.
When User Principal Name is not provided in user add request, adapter will generated the value for this attribute. (Please refer “Chapter 3. Active Directory Adapter user account management tasks->Adding user accounts->User principal name of a user account section” for more detail). Adapter then uses this generated User Principal Name to perform search in the forest. If adapter finds a user account with this generated User Principal Name it then appends a number, starting from 1, to the generated value to get a new User Principal Name.
Example,

Let’s consider that adapter has generated the User Principal Name as TestUser@MyDomain.com. If it is found to be used by an existing user account, adapter will then append the number 1 to generate the new value as TestUser1@MyDomain.com.

Now adapter will perform a search using this new value. If even the User Principal Name TestUser1@MyDomain.com is found to be used, again a new value is generated by appending number 2 as TestUser2@MyDomain.com. If TestUser2@MyDomain.com is also already in use, adapter will fail the user add request.

Note:

1. The User Principal Name’s search operation is costly in terms of adapter performance.
2. Due to replication delay adapter may not find a user with the current User Principal Name and will add the user account.
3. If there are two simultaneously running add request with the same User Principal Name, adapter may not find a user with the User Principal Name and both the user accounts will get added successfully.

It is recommended that you should have policy to generate unique User Principal Name on IBM Tivoli Identity Manager Server and should not rely on adapter to generate unique name.

Improve primary group lookup using cache

The reconciliation of primary group is costly operation and so a registry key “ReconPrimaryGroup” is used by adapter to improve reconciliation performance by setting this key to FALSE. This version of Windows Active Directory Adapter is improved to minimize primary group lookup searches using a local cache . If a set of users have the same primary group, then the lookup for that primary group will be done only once and not for each user account during a reconciliation operation.

Note: The primary group is cached for a particular reconciliation operation. If in your setup each user or most of users has different primary group then the benefit of this enhancements are nullified.
With this version the default value of ‘ReconPrimaryGroup’ is changed to FALSE’.

Adapter Version 5.0.7 Features

Support for Alert Processing on CN Attribute

IZ43288 - WIN-64 BIT WITH ALERT NON-COMPLIANCE POLICY ENFORCEMENT, WIN AD 64 BIT ADAPTER ENCOUNTERS ISSUES WITH CN.

When compliance alerts are enabled on IBM Tivoli Identity Manager, it is observed that the alerts keep alarming for user account class' 'cn' attribute. This is because of the fact that the attribute 'cn' in IBM Tivoli Identity Manager's schema is multi-valued where as the corresponding attribute on Active Directory 'cn' is single valued.

A new attribute is added in schema under erADAccount class with following details.
Attribute Name: erADFullName
OID: 1.3.6.1.4.1.6054.3.125.2.159
Description: Custom Common Name attribute
Data type: String
Custom Label: Full name

When the compliance alerts on Tivoli Identity Manager are enabled, avoid using the cn attribute on the account form. This issue may occur when alert non-compliance policy enforcement is set to automatic. No issue if compliance alerts on Tivoli Identity Manager is set as manual.
A new registry key "UseITIMCNAttribute" is introduced to the set of adapter registry keys. The default value of registry key UseITIMCNAttribute is TRUE. The adapter uses the registry key "UseITIMCNAttribute" to use either the cn or the erADFullName attribute.

When UseITIMCNAttribute = TRUE

- The adapter processes the IBM Tivoli Identity Manager's common schema attribute 'cn' for add, modify, and reconciliation operations.
- If the attribute 'erADFullName' is found in a request, this attribute will be failed by the adapter without considering the value.

When UseITIMCNAttribute = FALSE

- The adapter processes the erADFullName attribute for add, modify, and reconciliation operations.
- If the attribute 'cn' is found in a request, this attribute will be failed by the adapter without considering the value.

To use the erADFullName attribute on the account form, modify the profile using one of the following procedures

I. Modify the erADAccount.xml file of ADprofile.jar and importing the new profile on Tivoli Identity Manager

1. Copy the ADprofile.jar file to a temporary directory, example C:\Temp.
2. Extract the contents of ADprofile.jar file into the temporary directory by running the following command:
   cd C:\Temp
   jar -xvf ADprofile.jar
   The jar command creates the C:\Temp\ADprofile directory, which has all the profile files.
3. From the extracted ADprofile directory, open the erADAccount.xml file in a Text editor and make the following modifications and save the file:
   a. Replace the 'cn' attribute on account form with erADFullName attribute
   b. Change the attribute used to display on account form of the following attributes to erADFullName:
      erADManager
      erADEForwardTo
      erADEAllowedAddressList
      erADERstrctAdrsLs
      erADEDelegates

      For information about the required modification in erADAccount.xml file, see the table below. (Changes required are marked in blue)

Locate the following line(s) in erADAccount.xml file	Modification required to use erADFullName
<formElement direction="inherit" label="$cn"	<formElement direction="inherit"

name="data.cn"> <input type="text" size="50" name="data.cn"/>	label="$eradfullname" name="data.eradfullname"> <input type="text" size="50" name="data.eradfullname"/>
<formElement direction="inherit" label="$eradmanager" name="data.eradmanager"> <searchFilter type="input"> <filter>(&(objectclass=erADAccount)(eraddistinguishedname=*)(!(erisdeleted=Y))) </filter> <base>global</base> <attribute>cn</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>	<formElement direction="inherit" label="$eradmanager" name="data.eradmanager"> <searchFilter type="input"> <filter>(&(objectclass=erADAccount)(eraddistinguishedname=*)(!(erisdeleted=Y)))</filter> <base>global</base> <attribute>erADFullName</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>
<formElement direction="inherit" label="$eradeforwardto" name="data.eradeforwardto"> <searchFilter type="input"> <filter>(&(objectclass=erADAccount)(erADEAlias=*)(erADDistinguishedName=*)(!(erisdeleted=Y)))</filter> <base>global</base> <attribute>cn</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>	<formElement direction="inherit" label="$eradeforwardto" name="data.eradeforwardto"> <searchFilter type="input"> <filter>(&(objectclass=erADAccount)(erADEAlias=*)(erADDistinguishedName=*)(!(erisdeleted=Y)))</filter> <base>global</base> <attribute>erADFullName</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>
<formElement direction="inherit" label="$eradeallowedaddresslist" name="data.eradeallowedaddresslist"> <searchFilter multiple="true" type="select"> <filter>(&(objectclass=erADAccount)(erADEalias=*)(erADDistinguishedName=*)(!(erisdeleted=Y)))</filter> <base>global</base> <attribute>cn</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>	<formElement direction="inherit" label="$eradeallowedaddresslist" name="data.eradeallowedaddresslist"> <searchFilter multiple="true" type="select"> <filter>(&(objectclass=erADAccount)(erADEalias=*)(erADDistinguishedName=*)(!(erisdeleted=Y)))</filter> <base>global</base> <attribute>erADFullName</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>
<formElement direction="inherit" label="$eraderstrctadrsls"	<formElement direction="inherit" label="$eraderstrctadrsls"

name="data.eraderstrctadrsls"> <searchFilter multiple="true" type="select"> <filter>(&(objectclass=erADAccount)(erADEalias=*)(erADDistinguishedName=*)(!(erisdeleted=Y)))</filter> <base>global</base> <attribute>cn</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>	name="data.eraderstrctadrsls"> <searchFilter multiple="true" type="select"> <filter>(&(objectclass=erADAccount)(erADEalias=*)(erADDistinguishedName=*)(!(erisdeleted=Y)))</filter> <base>global</base> <attribute>erADFullName</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>
<formElement direction="inherit" label="$eradedelegates" name="data.eradedelegates"> <searchFilter multiple="true" type="select"> <filter>(&(objectclass=erADAccount)(erADEalias=*)(erADDistinguishedName=*)(!(erisdeleted=Y)))</filter> <base>global</base> <attribute>cn</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>	<formElement direction="inherit" label="$eradedelegates" name="data.eradedelegates"> <searchFilter multiple="true" type="select"> <filter>(&(objectclass=erADAccount)(erADEalias=*)(erADDistinguishedName=*)(!(erisdeleted=Y)))</filter> <base>global</base> <attribute>erADFullName</attribute> <sourceAttribute>erADDistinguishedName</sourceAttribute> <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADAccount</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>false</paginateResults> </searchFilter> </formElement>

4. Run the following command to create new jar file:
   cd C:\Temp
   jar -cvf ADprofile.jar ADprofile
   Note: The directory name and profile name is case sensitive, use the same case as above.
5. Import the new ADprofile.jar file on Tivoli Identity Manager.

II. Use "Form Customization" on Tivoli Identity Manager

1. Modify the account form of Windows Active Directory profile using Form Customization:
2. Remove the ‘cn’ attribute from the “User” tab
3. Add the ‘erADFullName’ attribute to the “User” tab
4. Change the attribute used to display on account form of the following attributes to erADFullName
   erADManager
   erADEForwardTo
   erADEAllowedAddressList
   erADERstrctAdrsLs
   erADEDelegates

Refer to the information center or the online help for information about using Form Customization

Unlocking WinAD Accounts without a Password Reset

MR0218091930 - unlock WinAD account without password reset.

The adapter functionality is enhanced to check the user account's lock status on Active Directory and accordingly succeed or fail account lock or unlock request. This helps in getting the account's status without use of reconciliation or event notification.
The adapter behaves as follows:

1. When a user lock request is submitted from IBM Tivoli Identity Manager and if the account is already locked on Active Directory, then the adapter will succeed the account lock request.
2. When a user lock request is submitted from IBM Tivoli Identity Manager and if the account is un-locked on Active Directory, then the adapter will fail the account lock request.
3. When a user unlock request is submitted from IBM Tivoli Identity Manager, then the adapter will succeed the account unlock request. This holds true regardless of whether the account is locked or unlocked on Active Directory.

Note: You can not lock account on Active Directory externally. Active Directory locks account after set number of failed logon attempts.

DAML Read Timeout Registry Key Option

MR0501091918 - ADK changes to DAML for adding timeout on read

From this version a new DAML protocol property "READ_TIMEOUT" is introduced to the list of DAML Protocol Properties on agentCfg utility. This applies to setups that have a firewall between IBM Tivoli Identity Manager and the adapter. This firewall has a time out value that is less than the maximum connection age DAML property on Tivoli Identity Manager. When your transactions run longer than the firewall's time out, the firewall terminates the connection. The sudden termination of connections might leave the adapter with incorrect connection threads causing the adapter to crash.
When the adapter crashes randomly because of the specified setup, change the value for the READ_TIMEOUT. **The value must be in seconds and not less than the firewall's time out value**. The default value is 0 seconds.

Follow the steps listed below to set non-zero seconds values to the "READ_TIMEOUT" DAML property

a) From the Start Menu, select Programs > Accessories > Command Prompt.

b) At the command prompt, change to the \bin directory for the adapter.
For example, type the following command, if the Active Directory Adapter is in the default location:
cd C:\Tivoli\Agents\ADAgent\bin

c) Type the following command and Enter configuration key for Agent:
agentCfg -agent ADAgent

d) From the Main Menu, select option B "Protocol Configuration".

e) At the Agent Protocol Configuration Menu, type C. The Configure Protocol Menu is displayed.

f) At the Configure Protocol Menu, type A. to select DAML protocol. The DAML Protocol Properties menu is displayed.

g) Type the letter 'K' from the menu option for "READ_TIMEOUT"

h) The following prompt is displayed:
Modify Property 'READ_TIMEOUT':

Type the time out value for Tivoli Identity Manager and the adapter connection in seconds.

Note: After you set a value for READ_TIMEOUT please restart the adapter service.

Support for "#" in Group Names

IZ52976 - WINAD GROUP MEMBERSHIP MODIFICATION FAILS IF GROUP CONTAINS # AND ADAPTER IS SET TO USE CN FOR GROUP

The adapter's reconciliation and event notification functionality is modified to return un-escaped value for erGroup attribute when the registry key UseGroup is set to CN

SearchTimeout Registry Key Option to Avoid AD Hang

PMR 34450,057,649 - RBC Recon Hang issue AD Adapter. New registry key "SearchTimeout" is added in adapter.

From this version of Windows Active Directory Adapter a new registry key "SearchTimeout" is added.
In some of the Active Directory setups, the adapter might not complete the reconciliation operation. This occurs when the Microsoft ADSI API GetNextRow halts indefinitely.
The adapter monitors the reconciliation operation. When you set this registry key to a non-zero value, the adapter process is terminated if there is no activity by the adapter in the reconciliation operation for the time in seconds specified in this key.
When you set the value of this registry key to 0 and if the adapter halts during the reconciliation operation, the reconciliation operation does not complete and the operation is timed out on Tivoli Identity Manager. In this case, restart the adapter service. The default value of the registry key is 0 seconds.

To set "SearchTimeout" registry key use agentcfg utility provided by Windows Active Directory Adapter.

Restricted Characters for erUID

Restricted the use of special characters in eruid attribute field on IBM Tivoli Identity Manager User account form.

From this version of the profile a new account form constraint "INVALID_CHARS" is added to the eruid attribute. This constraint will restrict the eruid attribute from having characters like / \ [ ] : ; | = , + * ? < > @ ". The restriction of these characters for eruid attribute comes from the restricted characters for samAccountName attribute on Active Directory, as the eruid attribute maps to samAccountName attribute. If you specify any of these characters for eruid attribute and submit the form, the following error message is displayed.
CTGIMU660E: A field contains characters that are not valid: invalid characters.

ADSI error 0x80004005 creating mailbox

MR050109660 - The error code 0x80004005 occurs because of replication delays or an exception in the Exchange libraries that the adapter uses. You can either submit the mailbox create request again after this failure or let the adapter retry to create mailbox by using two new registry keys "CreateMailboxRetryAttempts" and "CreateMailboxRetryDelay" from this version of Windows Active Directory Adapter.
Following section will explain how to use these registry keys.

CreateMailboxRetryAttempts:
The adapter uses this key when the Create a Mailbox operation fails with ADSI error code 0x80004005 – Unspecified Error. This key specifies the maximum number of retry attempts the adapter must make to create a mailbox. For example, if this key is set to 3 and the Create a Mailbox operation fails with error ADSI error code 0x80004005 – Unspecified Error the adapter performs maximum three retry attempts to create the mailbox. The default value is 0.

CreateMailboxRetryDelay:
The adapter uses this key when the Create a Mailbox operation fails with ADSI error code 0x80004005 – Unspecified Error. You can provide the number of seconds the adapter must wait before performing a retry. For example, when a CreateMailboxRetryAttempts is set to 3 and CreateMailboxRetryDelay is set to 5 and create mailbox fails with ADSI error code 0x80004005 – Unspecified Error, the adapter performs maximum three retry attempts to create the mailbox with 5 seconds delay between each retry attempt.

Note: The adapter uses this key only when CreateMailboxRetryAttempts is set to a non-zero retry attempt.

Adapter Version 5.0.10 Features

MR0210102732 - OCS support for AD adapter.

OCS support for users that are neither mail enabled nor have exchange mailboxes.

For OCS support Active Directory attribute erADEProxyAddresses needs to be updated. When the user account neither has a mailbox nor is mail enabled, then the adapter will fail modification to erADEProxyAddresses attribute.
To update proxyAddresses attribute on Active Directory make use of extended attributes using exschema.txt file. While setting extended attributes adapter does not check the mail status of the user account.
For example, add the following to exschema.txt file

erADExtendedProxyAddresses|proxyAddresses

This is an instruction to the adapter to manage Active Directory attribute “proxyAddresses” and to use “erADExtendedProxyAddresses” as the corresponding attribute on IBM Tivoli Identity Manager. Please refer to section “MR052609514 - customer wants to use the extend attribute transformed the name on itim side using the AD 5.0.5 adapter on ITIM 4.6 Server.” under “Configuration Notes” in Release Notes on how to specify extended attribute using exschema.txt file

Modify the Active Directory adapter profile, ADprofile.jar, to define a new attribute (erADExtendedProxyAddresses) and add it to erADAccount class in schema.dsml file. Please refer to “Chapter 7. Customizing the Active Directory adapter” in Configuration Notes for details on how to modify ADprofile.jar.

When a full reconciliation or user lookup is performed the values set for erADExtendedProxyAddresses attribute will also be returned for erADEProxyAddresses

attribute. This is because both erADExtendedProxyAddresses and erADEProxyAddresses correspond to the same attribute, proxyAddresses, on Active Directory.

Note:

- Adapter will not check for validity of values and their formats specified for Proxy Addresses through extended attribute.

MR0302105547 - Use erADLastLogonTimeStamp for AD dormant account report

In previous versions, Active Directory profile used attribute erADLastLogon for erLastAccessDate. The attribute erADLastLogon corresponds to lastLogon on Active Directory. The attribute lastLogon of a user account on Active Directory is updated only when the user logs in and is updated only on the DC against which the authentication happens. This attribute is not replicated amongst the DCs in the domain. Tivoli Identity Manager's dormant account report for Active Directory user accounts may not be accurate. This is because the adapter reads user accounts from a particular DC (specified as target server on service form or returned by DNS server as the nearest one) which can be different than the one using which the user's authentication has happened.

With this version, Active Directory profile will use attribute erADLastLogonTimeStamp for erLastAccessDate. The corresponding attribute on Active Directory, lastLogonTimestamp, is replicated across DC's. Using erADLastLogonTimeStamp attribute for erLastAccessDate will result in accurate dormant account reports.

To use erADLastLogon for erLastAccessDate will now require changes in resource.def file in ADProfile.jar

Replace following lines of resource.def file

```
<AttributeMap>
    <AttributeName="erLastAccessDate" Value="erADLastLogonTimeStamp"
Profile="account"/>
</AttributeMap>
```

With

```
<AttributeMap>
     <Attribute Name="erLastAccessDate" Value="erADLastLogon"
Profile="account"/>
</AttributeMap>
```

MR0226101912 - WinAD: Need a way to configure the length of the wait period for the retries of Win AD 64-bit Adapter for reconciliation.

A new registry key, ReconRetryWaitPeriod, is introduced to the set of adapter registry settings to support this enhancement. The default value for ReconRetryWaitPeriod is 300 (seconds)

In an organization, Active Directory always runs in cycles of low and heavy load depending on the number of authentication, replication, account management, and other Active Directory management requests. When Active Directory is observing heavy load and a reconciliation is performed, the adapter gets a special error message from Active Directory (Error_More_Data) indicating that it has not completed with reading all the user accounts and currently its observing a load. Upon this the adapter prints the incidence of this event to the log as "GetNextRow failed. Calling GetNextRow can potentially return more results. Provider: LDAP Provider". The adapter then waits for an interval specified by ReconRetryWaitPeriod registry key (in seconds) and retires again.

The Adapter is designed to retry the query three times before terminating the reconciliation. The adapter waits for a calculated time between each retry attempts, which is calculated as <RETRY_ATTEMPT_NUMBER> * <VALUE_OF_ ReconRetryWaitPeriod >
For example, when ReconRetryWaitPeriod is set to 40 seconds and the adapter receives Error_More_Data message from Active Directory. The wait period between each retry attempt is calculated as:
1 * 40 = 40 seconds – Before the first retry attempt;
2 * 40 = 80 seconds – Before the second retry attempt
3 * 40 = 120 seconds – Before the third (last) retry attempt.

Use agentCfg to set "ReconRetryWaitPeriod" to a value other than the default 300 seconds.

Note:

- The value for this registry should be numeric and in units of seconds.
- When SearchTimeout is also enabled, the value of ReconRetryWaitPeriod should be less than the value of SearchTimeout.
- The acceptable value for this key is from 0 to 214783647 seconds.

Setting Proxy Address:

Following table provides few scenarios to set proxy address. Please note that not all scenarios are covered in this table.

Sr. No	Scenario	Settings		Result	Comment
1	To set primary proxy address different than **Target Address** for a	i.	- **Auto Generate Email Address** is	The new proxy address will be set as	When **Auto Generate Email Address**

	Mail-enabled user.		TRUE. - Operation Type is REPLACE. -Add the **Target Address** as secondary proxy address by prefixing it with smtp. -Add the new proxy address to be set as primary with prefix SMTP.	primary proxy address and the **Target Address** will be set as secondary proxy address.	is TRUE, setting a primary proxy address different than Target Address can be achieved with a REPLACE operation type and adding Target Address value as secondary proxy address.
		ii.	- **Auto Generate Email Address** is FALSE. -Operation Type can be ADD or REPLACE. -Specify the new proxy address to be set as primary with prefix SMTP. -No need to set **Target Address** as secondary proxy address.	The new proxy address will be set as primary proxy address.	When **Auto Generate Email Address** is FALSE we can set primary proxy address using ADD operation Type, but any modifications or setting a new value can be done only with a REPLACE operation type.
2	To modify primary Proxy Address for a mailbox enabled user	i.	- Auto generate Email Address is set to TRUE. -Operation Type is REPLACE.	The new proxy address will be set as Primary proxy address.	-For a mailbox-enabled Exchange generates a primary proxy address using the Alias value. To set a different

			-Modify the primary proxy Address to new proxy address with prefix SMTP		primary proxy address we should modify it using REPLACE operation type.
		ii.	-Auto generate Email Address is set to FALSE. -Operation Type is ADD-DELETE. -Add a new proxy address with prefix SMTP.	The new proxy address will be set as Primary proxy address.	When Auto generate Email Address is FALSE, we can set a primary proxy address with ADD-DELETE operation type but, modification or setting a new value can be done only with a REPLACE operation type.

User Exchange attributes, erADESMTPEmail and erADEX400Email.

erADESMTPEmail
The erADEXMTPEmail attribute holds the primary SMTP proxy address set for an user account. When the value of this attribute is modified, the new value will be used as the primary SMTP address.

erADEX400Email
This attribute is not managed by the adapter. If exist in a request, it will be ignored. The adapter will reconcile value(s) of this attribute.

MR052609514 - customer wants to use the extend attribute transformed the name on itim side using the AD 5.0.5 adapter on ITIM 4.6 Server.

The adapter now supports mapping of attribute names for extended attributes. With this enhancement a different attribute name can be used on IBM Tivoli Identity Manager than the attribute name on Active Directory.

As before the exschema.txt file will be used to specify the extended attributes. If required to use a different attribute name on IBM Tivoli Identity Manager than the attribute name on Active Directory, specify the attribute name in exschema.txt file in the following format:
Attribute name on IBM Tivoli Identity Manager followed by a pipe '|' and then followed by the attribute name on Active Directory.
For example,

```
erADUserInfo|info
```

As before there should be only one attribute on each line.
If you wish to use the same attribute name on IBM Tivoli Identity Manager as on Active Directory, then use the existing format i.e. just give the Active Directory side attribute name.
For example,

```
Info
```

Or

```
Info|Info
```

Both the above forms are valid, use either one.

Note:

- When you use a different attribute name for IBM Tivoli Identity Manager ensure that the attribute name rules, as defined by the Directory Server used, are followed.
- The attribute name must not have a '|' in the attribute name.

Enable/Disable "UseThreadPooling"

There is an issue In the Microsoft CDOEXM library used by Windows Active Directory Adapter to perform Exchange tasks. A ticket was also opened with Microsoft, Case ID "SRZ080104000181", for the same.
Agent is redesigned as described in Microsoft Case ID. Adapter now implements thread pool. A predefined number of threads (12) are created at the start of adapter and are used to perform all operations. These threads will be destroyed only at the end i.e. when adapter itself is stopped.
A new registry key "UseThreadPooling" is introduced. By default this key is set to "FALSE" so that existing customers are not affected.
When UseThreadPooling is set to TRUE Thread Pooling is enabled, with all the threads initialized at the start of Agent Service and uninitialized when the Agent service stops.
When UseThreadPooling is set to FALSE Thread Pooling is disabled. In this scenario threads will be created and destroyed on per request.

Thread Pooling can be used in the following scenarios:

1. If you are experiencing high memory usage then set this key to "TRUE".
2. If you are experiencing the following error message during the Exchange related operations.
   errorMessage="Unable to contact Exchange services. ADSI Result code: 0x80004002"

Adapter Version 5.0.11 Features

MR0204103013 - AD adapter support for DNWithBinary.

Adapter is enhanced to support *DNWithBinary* syntax for extended attribute.

With this enhancement adapter is now able to perform add, delete, modify, recon operations on the extended attributes of type *DNWithBinary*.

Note:

1) Adapter based filtering is not supported for syntax *DNWithBinary* for extended attribute.
2) Adapter based event notification has some limitations for syntax *DNWithBinary*. Refer *Known Issues* section**.**

The *DNWithBinary* attribute store values in following format in Active Directory:

B :< char count> :< binary value> :< object DN>
Here "<char count>" is the number of hexadecimal digits in "<binary value>"
"<binary value>" is the hexadecimal representation of the binary value and
"<object DN>" is a distinguished name of existing user object.

To set the extended attributes of type *DNWithBinary* on Active Directory you need to specify the value of attribute only in the above given format.

Example:

- If you need to set the attribute **msRTCSIP-UserPolicy**, value could be:
  B:8:01000000:CN={FCE1E52A-59D1-4FBB-9CB5-2679247F7943},CN=Policies,CN=RTC Service,CN=Services,CN=Configuration,DC=evaluation,DC=test

- If you need to set the attribute **otherWellknowObjects**, value could be:
  B:32:df447b5eaa5b11d28d5300c04f79ab81:CN=User01,OU=Testorg,DC=pwdtest,DC=COM

Adapter will check for the validity of formats specified for extended attributes of type *DNWithBinary*.

Additions to the User Guide

Extended Attributes

The adapter supports processing of multi valued string syntaxes extended attribute as add, delete, and replace attribute operation. This signifies whether to append, delete, or replace values in the request to/from the set of values set for the corresponding Active Directory side attribute.

Please refer to section “MR052609514 - customer wants to use the extend attribute transformed the name on itim side using the AD 5.0.5 adapter on ITIM 4.6 Server.” under “Configuration Notes” in Release Notes on how to specify extended attribute using exschema.txt file.

There can be cases where you have an extended attribute with a corresponding Active Directory side attribute which is already managed by the adapter. When a full reconciliation or user lookup is performed the values set using the extended attribute will also be returned for the attribute which is already been managed by the adapter and vice versa. This is because both these attributes corresponds to the same attribute on Active Directory.

Please refer to “Appendix B. Active Directory Adapter attributes” in User Guide for list of adapter attributes and their corresponding Active Directory side attribute name which the adapter manages.

Chapter 4. Troubleshooting the Active Directory Adapter errors- > Active Directory Adapter errors (For MR0204103013 - AD adapter support for DNWithBinary)

No.	Error Messages	Recommended action
1	“Value specified is not in the proper format”	Ensure that value format of extended attribute of type DNWithBinary is **B :< char count> :< binary value> :< object DN>**
2	"Value specified for the attribute does not start with character 'B'."	Ensure that value specified for extended attribute of type DNWithBinary is start with character ‘**B**’ only.
3	"Value given after '**B:**' is not correct. Expected value is the total number of Hexadecimal Digit count."	For extended attribute of type DNWithBinary, verify that value given after **B:** i.e. **<char count>**is total number of Hexadecimal Digit count. It should not contain any alphabetical character or any special character.
4	"Hexadecimal value does not contain the number of characters specified in the character count."	For extended attribute of type DNWithBinary, verify that total number of hexadecimal digit count specified in the **<char count >** is equal to number of hexadecimal characters specified in the **<binary value>**
5	"Wrong Digit in Hex String"	For extended attribute of type DNWithBinary, verify that value given in the **<binary value>** contain only hexadecimal character i.e. it should contain characters 0-9 or A,B,C,D,E,F or combination of both.
6	"value is not set on resource due to invalid constraint"	This error occurs when the specified value for the extended attribute of type DNWithBinary

		violates any constraint associated with that attribute. **For example**, a constraint could be: 1)**<object DN>** in the value should be a distinguished name of existing user object 2) Maximum or minimum number of bits in the hexadecimal value. Ensure that the specified value for the attribute does not violate these constraints.
7.	"Hexadecimal value should always contain even number of characters."	For extended attribute of type DNWithBinary, verify that value given in the **<binary value>** contain only even number of hexadecimal characters.

Customizing or Extending Adapter Features

The Identity Manager adapters can be customized and/or extended. The type and method of this customization may vary from adapter to adapter.

Getting Started

Customizing and extending adapters requires a number of additional skills. The developer must be familiar with the following concepts and skills prior to beginning the modifications:

- LDAP schema management
- Working knowledge of scripting language appropriate for the installation platform
- Working knowledge of LDAP object classes and attributes
- Working knowledge of XML document structure

Note: This adapter supports customization both through the use of pre-Exec and post-Exec scripting and schema extensions using the extshema.txt file.

Tivoli Identity Manager Resources:

Check the “Learn” section of the Tivoli Identity Manager Support web site for links to training, publications, and demos.

Support for Customized Adapters

The integration to the Identity Manager server – the adapter framework – is supported. However, IBM does not support the customizations, scripts, or other modifications. If you experience a problem with a customized adapter, IBM Support may require the problem to be demonstrated on the GA version of the adapter before a PMR is opened.

Supported Configurations

Installation Platform

The IBM Tivoli Identity Manager Adapter was built and tested on the following product versions.

Adapter Installation Platform:

Windows 2003	Standard Edition 32-bit or 64-bit OS on x86 and x64 compatible CPU
Windows 2003	Enterprise Edition 32-bit or 64-bit OS on x86 and x64 compatible CPU
Windows 2008	Standard Edition 32-bit or 64-bit OS on x86 and x64 compatible CPU
Windows 2008	Enterprise Edition 32-bit or 64-bit OS on x86 and x64 compatible CPU
Windows 2008 R2	Standard Edition 32-bit or 64-bit OS on x86 and x64 compatible CPU
Windows 2008 R2	Enterprise Edition 32-bit or 64-bit OS on x86 and x64 compatible CPU
Windows 2008 R2 Core	Enterprise 64-bit OS on x64 compatible CPU

Managed Resource:

Active Directory on Windows 2003 Standard or Enterprise Edition 32-bit or 64-bit OS
Active Directory on Windows 2008 Standard or Enterprise Edition 32-bit or 64-bit OS
Active Directory on Windows 2008 R2 Enterprise Edition 64-bit OS
Active Directory on Windows 2008 R2 Core Enterprise 64-bit OS on x64 compatible CPU

With optional:
Exchange 2003 Server Standard or Enterprise Edition
-- with --
Exchange Administration Tools 2003 with SP2

Note: Microsoft supports Exchange 2003 only on 32-bit versions of Windows 2003. Windows 2003 64-bit and Windows 2008 are not supported. See Microsoft product documentation for more information.

IBM Tivoli Identity Manager:

Identity Manager v5.0

NOTE: Exchange 2007 is not supported by this adapter. Use the Windows Active Directory x64 Adapter to manage Exchange 2007.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.
IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

```
IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY  10504-1785  U.S.A.
```

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

```
IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan
```

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged should contact:

```
IBM Corporation
2ZA4/101
11400 Burnet Road
Austin, TX 78758  U.S.A.
```

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Trademarks

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM
IBM logo
Tivoli

Adobe, Acrobat, Portable Document Format (PDF), and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Cell Broadband Engine and Cell/B.E. are trademarks of Sony Computer Entertainment, Inc., in the United States, other countries, or both and is used under license therefrom.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT®, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel®, Intel logo, Intel Inside®, Intel Inside logo, Intel Centrino™, Intel Centrino logo, Celeron®, Intel Xeon™, Intel SpeedStep®, Itanium®, and Pentium® are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the U.S., other countries, or both.

ITIL® is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

End of Release Notes