APPENDIX D

Troubleshooting Active Directory Agent Issues

This appendix contains information that would help you identify and resolve issues that you might experience while using the AD Agent. This appendix contains the following sections:

- Obtaining Troubleshooting Information, page D-1
- Enabling Internal Debug Logs in the AD Agent, page D-2
- Configuration Issues, page D-4

Obtaining Troubleshooting Information

You can obtain troubleshooting information from the official customer logs that are generated by the AD Agent. These logs are available locally on the AD Agent machine in the following directory: **C:\IBF\radiusServer\runtime\logs\localStore**.

You can also send the AD Agent's customer logs to a Syslog server. See "Configuring AD Agent to Send Logs to a Syslog Server" section on page 2-9 for information on how to configure a Syslog server to receive these logs.

You can use the **adacfg options set -logLevel** command to control the level of detail in the customer logs, both for localStore and syslogs.

See Appendix B, "Customer Log Messages," for a list of relevant Customer Log messages.

By default, the logging level is set to INFO and only informational messages will be reported. If you are attempting to troubleshoot a specific problem, you can change this logging level to obtain additional information. The valid options for the logging level, in order of decreasing verbosity, are:

- DEBUG
- INFO
- WARN
- ERROR
- FATAL

See "adacfg options set" section on page A-11 for more information about this command.

In addition to the Customer Logs, another source of information that might be helpful to you during troubleshooting is the Windows Application Event Log, which you can view using the Windows Event Viewer tool (**Event Viewer (Local) > Applications and Services Log > Cisco AD Agent**). This tool records the start and stop events of the AD Agent software and the internal "AD Observer" and "RADIUS Server" processes. See Appendix C, "Windows Application Event Logs," for more information.

Finally, when reporting problems, you might be asked to also enable the internal debug logs on your AD Agent machine, and to send these logs in addition to the Customer Logs. These logs can assist in the diagnosis and resolution of your problems. To enable these internal debug logs, see "Enabling Internal Debug Logs in the AD Agent" section on page D-2.

Enabling Internal Debug Logs in the AD Agent

For advanced troubleshooting, there are two types of internal debug logs that you can enable:

- AD Observer Logs, page D-2
- RADIUS Server Logs, page D-3

AD Observer Logs

The "C:\IBF\adObserver\logconfig.ini" file specifies the internal debug logging level for the AD Observer subcomponent. By default, the LOG_LEVEL is set to LOG_NONE, which indicates that no internal debug log would be generated for the AD Observer subcomponent.

The LOG_LEVEL can take any one of the following values:

- LOG_VERBOSE—Most verbose logs
- LOG_DEBUG—Contains troubleshooting and debug information
- LOG_INFO—Contains informational messages
- LOG_WARN—Contains warning messages
- LOG_ERROR—Contains error messages
- LOG_FATAL—Contains only fatal error messages

The log levels have decreasing level of information in them with the LOG_VERBOSE containing maximum information and LOG_FATAL containing the least amount of information. We recommend that you choose the LOG_DEBUG to obtain troubleshooting information.

To enable the AD Observer internal debug log:

Step 1 From your AD Agent machine, go to the **C:\IBF\adObserver directory**.

Step 2 Use any text editor, such as Notepad to open the **logconfig.ini** file.

Step 3 Modify the last sentence of this configuration file as follows:

LOG_LEVEL=LOG_VERBOSE

Step 4 Save the **logconfig.ini** file.

Step 5 Restart the AD Agent using the **adactrl restart** command for this change to take effect.

You have enabled the AD Observer internal debug log. The AD Agent will generate the ADObserverLog.txt file in the following directory: **C:\IBF\adObserver**.

Note This "LOG_LEVEL" setting for the internal debug mechanism of the AD Observer subcomponent is not related to the -logLevel configurable option of the **adacfg options set** command.

RADIUS Server Logs

The RADIUS server runtime debug log configuration file allows you to enable or disable internal debug logging for the various RADIUS server subcomponents. This file is available in the following location: **C:\IBF\radiusServer\runtime\win32\config\RuntimeDebugLog.config**.

By default, the debug logging is disabled for all subcomponents.

In this configuration file, sentences of the following form, list the RADIUS server subcomponents for which debug logging can be turned off or on:

#components.[Acs.RT.]*variable*=off

where *variable* could be any one of the following subcomponents:

- ConfigVersionManager
- ConfigManager.XmlManager
- Statistics
- ConfigManager
- Logging
- Dictionary
- MessageCatalog.CatalogRepository
- Crypto.CRLHttpWorker
- EventHandler
- EventHandler.EventDispatchTable

To enable internal debug logging for any of the RADIUS server subcomponents:

Step 1 From your AD Agent machine, go to the **C:\IBF\radiusServer\runtime\win32\config**.

Step 2 Use any text editor, such as WordPad, open the **RuntimeDebugLog.config** file.

Step 3 In this configuration file, at the end of the line that lists the RADIUS server subcomponent whose debug logs you want to enable, replace the word **off** with **on**.

Replace the word **off** with **on** for all the subcomponents whose debug logs you want to enable.

Note This value is case sensitive. Use lower case letters for the words **off** and **on**.

Step 4 Save the **RuntimeDebugLog.config** file.

The AD Agent detects the change in the RADIUS server configuration file automatically and the RADIUS server debug logs are enabled. These logs will be available in the following location:

C:\IBF\radiusServer\runtime\logs\radiusServer_debug.log.

Configuration Issues

This section lists some of the commonly observed configuration issues. This section contains the following topics:

- Requests from Client Device are Seemingly Ignored, page D-4
- The adacfg client status Command Reports Client Device as Being "Out-of-Sync" for Unclear Reasons, page D-5
- IP-to-user-identity Mappings Disappear Too Quickly from the AD Agent Cache, page D-5
- User Logons Authenticated By a Given DC Machine Not Being Detected (and Acted Upon) by AD Agent, page D-6
- The 'adacfg dc list' Command Shows Domain Controller Machine Has Not Reached the 'up' State, page D-7
- AD Agent Does Not Function At All, page D-8
- The 'adacfg dc list' Command Shows Domain Controller Machine Has Reached the "down(no-retry)" State, page D-8
- Rebooting AD Agent Machine Leads to Logon Failure, page D-9

Requests from Client Device are Seemingly Ignored

Symptoms or Issue	Requests from the client device do not reach the AD Agent machine, or seem to be ignored.
Possible Causes	1. The client device might not be properly configured to interact with the AD Agent machine. 2. Windows Firewall might block the RADIUS traffic. 3. A wrong RADIUS shared secret was entered when the **adacfg client create** command was used to configure the client device on the AD Agent machine.
Resolution	1. Ensure that the client device is properly configured to interact with the AD Agent machine. 2. Ensure that if Windows Firewall, or a similar firewall software is running, then the necessary firewall exceptions have been configured, as described in "Connectivity Requirements" section on page 2-2. 3. Check the Customer Logs (localStore or syslogs) for the presence of log messages that report an invalid RADIUS Authenticator field or Message-Authenticator attribute. If such messages have been found, then ensure that the client device and the AD Agent have both been correctly configured to use the same RADIUS shared secret.

The adacfg client status Command Reports Client Device as Being "Out-of-Sync" for Unclear Reasons

Symptoms or Issue	After the AD Agent machine sends notification updates to the client device through a RADIUS CoA-Request, it is not successfully receiving a CoA-ACK from the client device.
Possible Causes	1. The client device might not be sending the RADIUS CoA-ACK at all because it is currently down, or not configured properly. 2. The Windows Firewall might block the RADIUS traffic. 3. The client device is sending the CoA-ACK, but the AD Agent machine is dropping it because of a wrong RADIUS shared secret.
Resolution	1. Ensure that the client device is currently up, and properly configured to interact with the AD Agent machine. 2. Ensure that if the Windows Firewall, or a similar firewall software is running, then the necessary firewall exceptions have been configured, as described in "Connectivity Requirements" section on page 2-2. 3. Check the Customer Logs (localStore or syslogs) for the presence of log messages that report an invalid RADIUS Authenticator field or Message-Authenticator attribute. If such messages have been found, then ensure that the client device and the AD Agent have both been correctly configured to use the same RADIUS shared secret.

IP-to-user-identity Mappings Disappear Too Quickly from the AD Agent Cache

Symptoms or Issue	The IP-to-user-identity mappings disappear too quickly from the AD Agent cache.
Possible Causes	The time period corresponding to the current setting of the 'userLogonTTL' configurable option in the **adacfg options set** command is too short.
Resolution	Configure a longer time period for the user logon TTL. See adacfg options set, page A-11 for more information.

User Logons Authenticated By a Given DC Machine Not Being Detected (and Acted Upon) by AD Agent

Symptoms or Issue	User logons authenticated by a given DC machine not being detected (and acted upon) by the AD Agent.
Possible Causes	1. The given DC machine might not be properly patched, causing authentication events to sometimes not be written to its Security Log. 2. The Audit Policy on that DC machine might not be properly configured. 3. The AD Agent might be detecting the mapping-update, but then dropping it for one of several reasons (as reported in the Customer Logs). One possible reason, for example, might be "mapping-update having timestamp in future," which can happen if the clock of the DC machine is more than 10 minutes ahead of the clock of the AD Agent machine.
Resolution	1. Ensure that the given DC machine is properly patched. 2. Ensure that the Audit Policy on the given DC machine is properly configured. 3. Use the localStore repository on the AD Agent machine (or the syslogs) to ensure that the AD Agent machine is receiving the corresponding mapping-update, and not dropping it for any reason. If the AD Agent machine is dropping the mapping-update for whatever reason, ensure that this problem is corrected. For example, if mapping-updates have a "timestamp in future," ensure that the clocks of the DC machine and the AD Agent machine are properly synchronized.

The 'adacfg dc list' Command Shows Domain Controller Machine Has Not Reached the 'up' State

Symptoms or Issue	The **adacfg dc list** command shows that the domain controller machine has not reached the "up" state.
Possible Causes	1. The domain controller machine might not be running a supported version of Windows Server. 2. The domain controller machine might not be properly patched. 3. The Windows Firewall, or a similar firewall software, might be blocking WMI traffic between the domain controller machine and the AD Agent machine. 4. The AD Agent machine might not be joined to an AD domain, or proper trust relationship might not exist between the AD domain of the domain controller machine and the AD domain to which the AD Agent machine is joined. 5. The values entered in the adacfg dc create command might be incorrect. Specifically, you might not have entered the full DNS name of the domain, or the account credentials were incorrect, or the account might have insufficient privileges to read the Security Log of the domain controller machine.
Resolution	1. Ensure that the domain controller machine is running a supported version of the Windows Server, and that it is properly patched. 2. Ensure that if Windows Firewall, or a similar firewall software is running, then the necessary WMI exceptions have been properly configured. 3. Ensure that the AD Agent machine is joined to an AD domain that has a proper trust relationship with the AD domain of the domain controller machine. 4. Ensure that the values entered in the adacfg dc create command are correct. Specifically, ensure that you provide the full DNS name of the domain and credentials for an account that has sufficient privileges to read the Security Log of the domain controller machine. 5. If necessary, enable the internal debug log of the AD Observer subcomponent, and check for the following: – The RPC server is unavailable (0x800706ba)—If you see this message, it suggests that either the domain controller machine is down, or communication is blocked because of the use of a firewall without the necessary exceptions. – Access is Denied (0x80070005)—If you see this message, it suggests that the specified account does not have sufficient privileges to read the Security Log of the domain controller machine, or that the credentials are wrong.

AD Agent Does Not Function At All

Symptoms or Issue	AD Agent is not functioning at all and when you enter various CLI commands, you consistently get the error message, `Couldn't connect to server!`.
Possible Causes	A few antivirus software programs are known to block cygwin1.dll as a virtualization-related threat. However, this report should be treated as false-positive. AD Agent does not contain any malware.
Resolution	1. After you run the AD Agent installer executable, check the logs of your antivirus software to see if it blocked C:\IBF\radiusServer\cygwin\bin\cygwin1.dll (or any other items under the C:\IBF folder) as a potential threat. 2. If any such AD Agent subcomponents were blocked, configure your antivirus software to explicitly allow them to run unblocked.

The 'adacfg dc list' Command Shows Domain Controller Machine Has Reached the "down(no-retry)" State

Symptoms or Issue	The **adacfg dc list** command shows that the domain controller machine has reached the "down(no-retry)" state.
Possible Causes	The WMI service on the domain controller machine might be unresponsive because the domain controller machine might not be properly patched.
Resolution	1. Ensure that the domain controller machine is properly patched. 2. Try to restart the WMI service on the domain controller machine. 3. To force the AD Agent to retry the connection, do one of the following: – Delete and re-create the domain controller configuration using the **adacfg dc create** command. – Restart the AD Agent using the **adactrl restart** command.

Rebooting AD Agent Machine Leads to Logon Failure

Symptoms or Issue	Rebooting the AD Agent machine leads to the following error: `Windows could not start the Cisco AD Agent service on Local Computer. ERROR 1069: The service did not start due to a logon failure`

Possible Causes	The AD Agent might have been directly installed on more than one domain controller machine (for the same AD domain).
	In such a case, during AD Agent installation, you would have encountered a dialog box with the following message:
	`'IBF_SERVICE_USER' account already exists. OK to recreate? (Pressing 'No' will abort the installation.)`
	`WARNING: Make sure you are NOT attempting to install AD Agent directly on more than one DC machine (for the same AD domain)!`
	You might have chosen 'Yes,' which would lead to this problem.

Resolution	1. Manually set the password of the non-local account "IBF_SERVICE_USER," created on the domain, to a known value. 2. Manually modify the "Cisco AD Agent" item in the "Services" panel of each domain controller machine where the AD Agent was installed to use this new password, and then restart this service or reboot the domain controller machine.