Installation and Setup Guide for the Cisco Active Directory Agent, Release 1.0

August 13, 2011

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-25134-01

Installation and Setup Guide for the Cisco Active Directory Agent, Release 1.0

CONTENTS

Preface

Revised: August 2, 2011, OL-25134-01

This guide describes how to install the Cisco Active Directory Agent and configure it. Throughout this guide, the term AD Agent is used to refer to the Cisco Active Directory Agent.

This preface covers the following topics:

Audience

This guide is written for network administrators who will be using the Active Directory Agent in their deployments. This guide assumes you have a working knowledge of networking principles and applications, and have experience as a network system administrator.

Document Organization Map

The topics in this guide are organized in the following way:

- Preface
- Overview of the Cisco Active Directory Agent
- Installing and Configuring Active Directory Agent
- Active Directory Agent Command Reference
- Customer Log Messages
- Windows Application Event Logs
- Troubleshooting Active Directory Agent Issues

Document Conventions

This guide uses the convention whereby the symbol ^ represents the key labeled *Control*. For example, the key combination **^z** means hold down the **Control** key while you press the **z** key.

Command descriptions use these conventions:

- Examples that contain system prompts denote interactive sessions, indicating the commands that you should enter at the prompt. The system prompt indicates the current level of the EXEC command interpreter. For example, the prompt `Router>` indicates that you should be at the *user* level, and the prompt `Router#` indicates that you should be at the *privileged* level. Access to the privileged level usually requires a password.
- Commands and keywords are in **boldface** font.
- Arguments for which you supply values are in *italic* font.
- Elements in square brackets ([ ]) are optional.
- Alternative keywords of which you must choose one are grouped in braces ({ }) and separated by vertical bars (|).

Examples use these conventions:

- Terminal sessions and sample console screen displays are in `screen` font.
- Information you enter is in **`boldface screen`** font.
- Nonprinting characters, such as passwords, are in angle brackets (< >).
- Default responses to system prompts are in square brackets ([]).
- An exclamation point (!) at the beginning of a line indicates a comment line.

Caution Means *reader be careful*. You are capable of doing something that might result in equipment damage or loss of data.

Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.

Note Means *reader take note*. Notes identify important information that you should reflect upon before continuing, contain helpful suggestions, or provide references to materials not contained in the document.

Documentation Updates

Table 1 Updates to the Installation and Setup Guide for the Active Directory Agent

Date	Description
June 23, 2011	Added the note" Internationalization is not supported." in chapter 2.
June 13, 2011	Cisco AD Agent, Release 1.0

Related Documentation

Release-Specific Documentation

Table 2 lists the product documentation available for the AD Agent, Release 1.0.

Table 2 Release-Specific Documentation

Document Title	Location
Installation and Setup Guide for the Cisco Active Directory Agent, Release 1.0	http://www.cisco.com/en/US/docs/security/ibf/setup_guide/ad_agent_setup_guide.html
Release Notes for the Cisco Active Directory Agent, Release 1.0	http://www.cisco.com/en/US/docs/security/ibf/release_notes/ibf10_rn.html
Open Source Used in Cisco Active Directory Agent 1.0	http://www.cisco.com/en/US/docs/security/ibf/open_source_license_document/ipcentral.pdf

Other Related Documentation

Links to Adaptive Security Appliance (ASA) 5500 Series Release 8.4.2 documentation and Cisco IronPort Web Security Appliance (WSA) documentation are available on Cisco.com at the following locations:

- Cisco ASA 5500 Series Adaptive Security Appliances Page

  http://www.cisco.com/en/US/products/ps6120/tsd_products_support_series_home.html
- Cisco IronPort Web Security Appliances Page

  http://www.cisco.com/en/US/products/ps10164/tsd_products_support_series_home.html

Notices

See http://www.cisco.com/en/US/docs/security/ibf/open_source_license_document/ipcentral.pdf for all the Open Source Licenses used in Active Directory Agent, Release 1.0.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as an RSS feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service. Cisco currently supports RSS Version 2.0.

CHAPTER 1

Overview of the Cisco Active Directory Agent

The Cisco Active Directory Agent (AD Agent) is a component that runs on a Windows machine; monitors in real time a collection of Active Directory domain controller (DC) machines for authentication-related events that generally indicate user logins; learns, analyzes, and caches mappings of IP addresses and user identities in its database; and makes the latest mappings available to its client devices.

Client devices, such as the Cisco Adaptive Security Appliance (ASA) and the Cisco IronPort Web Security Appliance (WSA), interact with the AD Agent using the RADIUS protocol in order to obtain the latest set of IP-to-user-identity mappings, in any one of the following ways:

- On-Demand—The AD Agent can respond to an on-demand query from the client device for a specific mapping.
- Bulk Download—The AD Agent can respond to a request from the client device for the entire set of mappings currently in its cache.

For both the on-demand and bulk-download methods, the request from the client device can be specially tagged to indicate that it also includes a request for notification regarding any subsequent updates.

For example, when a client device requests a basic on-demand query, the AD Agent will respond with the specific mapping that might have been found in its cache, and does not send any further updates about that mapping. On the other hand, if the on-demand query also includes a request for notification, the initial response from the AD Agent will be the same as before and if, at a later point in time, that specific mapping undergoes a change, then the AD Agent will proactively notify the requesting client device (as well as any other client devices that have registered for notification) about the change in that specific mapping.

Similarly, when a client device requests a basic bulk download, the AD Agent will transfer a snapshot of the session data containing all of the mappings currently found in its cache, and does not send any further updates. On the other hand, if the request is to register for replication, then the initial response from the AD Agent will be the same as before, and if, at a later point in time, the set of mappings undergoes any sort of change (new mappings added or certain mappings changed and so on), then the AD Agent will proactively notify the requesting client device (as well as any other client devices that have registered for replication) about these changes, relative to the snapshot that was previously sent.

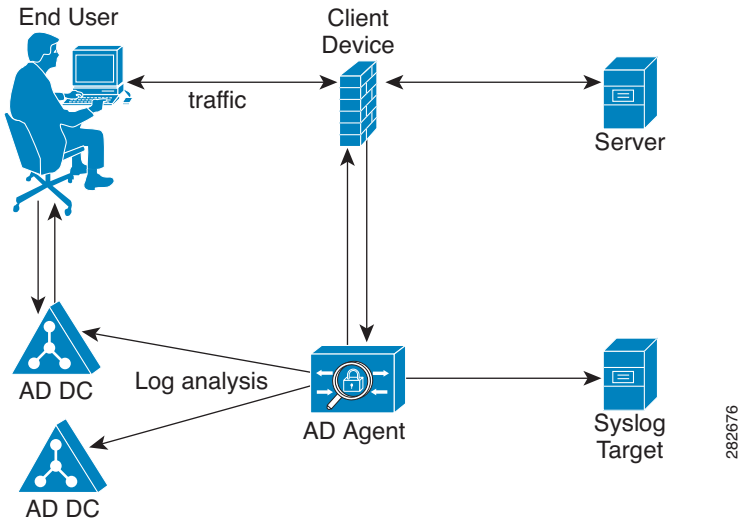
The IP-to-user-identity mappings that are discovered, maintained, and provided by the AD Agent can include not only IPv4 addresses, but also IPv6 addresses.

The AD Agent can send logs to one or more syslog servers.

The AD Agent will continue to function if any of the AD domain controllers or the client devices have failed. It will obtain information from other domain controllers. However, there is no failover for the AD Agent. The Cisco AD Agent internally contains a “watchdog” functionality that continuously monitors the Windows processes internal to the AD Agent, automatically restarting them if it detects that they have crashed.

Figure 1-1 illustrates the role of AD Agent in an example scenario.

Figure 1-1 AD Agent Used in a Solution



In this example, a user logs in from a computer and generates web traffic by requesting access to a server. The client device intercepts the web traffic and sends a RADIUS request to the AD Agent asking for the user who logged into the computer. The AD Agent, which has been maintaining the latest set of IP-to-user-identity mappings, sends the user information to the client device. The client device uses the the user identity information to determine whether or not to grant access to the end user.

The AD Agent interacts with the following components in a network:

- Client Devices
- Active Directory Domain Controller Machines
- Syslog Servers

Note AD Agent can support up to 100 client devices and 30 domain controller machines, and can internally cache up to 64,000 IP-to-user-identity mappings.

Client Devices

Client devices are responsible for actively retrieving (and/or passively receiving) the latest IP-to-user-identity mappings from the AD Agent.

A client device can retrieve these mappings from the AD Agent in the following ways:

- Query the AD Agent for each new IP
- Maintain a local copy of the entire user identity and IP address database

In addition to receiving the latest set of IP-to-user-identity mappings from the AD Agent, a client device can also send updates of mappings that it learns through other mechanisms to the AD Agent. For example, the ASA device updates the AD Agent with:

- new mappings learned during web authentication fallback (for IP addresses that the AD Agent could not map to user-identity)
- new mappings learned from VPN sessions
- mapping-removals associated with logoffs or disconnects learned from VPN/Cut-Through Proxy or through NetBIOS probing or MAC checking

These updates are sent as RADIUS Accounting-Request messages.

Note For information about configuring ASA devices to send notifications to the AD Agent, refer to the ASA end-user documentation.

Active Directory Domain Controller Machines

Though Active Directory is part of this solution, it is managed by Active Directory administrators. The reliability and accuracy of the data depends on the Active Directory domain controller's data. The AD Agent monitors, learns, and reads events from Active Directory domain controllers.

The AD Agent only monitors authentication events in which Kerberos is used to authenticate the user.

The events that the AD Agent monitors are usually triggered by logins, but can also be triggered by other activities such as:

- The use of the "runas" Windows command
- The use of the "net user" Windows command

The AD Agent is able to monitor up to 30 Active Directory domain controller machines, each running one of the following supported versions of Windows Server:

- Windows Server 2003
- Windows Server 2008
- Windows Server 2008 R2

Note Windows Server 2003 R2 is not supported.

It is important to verify that each and every Active Directory domain controller machine running Windows Server 2008 or Windows Server 2008 R2 has the appropriate Microsoft hotfixes installed, as detailed in "Active Directory Requirements" section on page 2-5. You must apply the hotfixes regardless of whether the AD Agent is installed directly on the domain controller machine, or monitoring the domain controller machine remotely.

Similarly, it is important to verify that the Audit Policy on each Active Directory domain controller machine allows for auditing of successful authentication attempts, as detailed in "Configuring AD Agent to Obtain Information from AD Domain Controllers" section on page 2-9.

The AD Agent can monitor domains that have a trust relationship with the domain to which the AD Agent machine is joined. The AD Agent supports the following Active Directory structures:

- Single forest, single domain
- Single forest, multiple domains
- Multiple forests

Syslog Servers

The AD Agent can forward logs containing administrative and troubleshooting information to one or more syslog servers. The contents of these logs are identical to that of the customer logs that are locally available on the AD Agent machine, in the C:\IBF\radiusServer\runtime\logs\localStore\ directory. The syslog mechanism allows this information to be distributed remotely, to any target machine running a syslog server and capable of receiving syslog messages.

CHAPTER 2

Installing and Configuring Active Directory Agent

Active Directory Agent is a software application that comes packaged as a Windows installer. You must install it on a Windows machine and configure it with client devices and AD domain controllers.

This chapter contains the following topics:

Note You must refer to the ASA end-user documentation for configurations related to the ASA device.

Requirements

This section contains the following topics:

Hardware Requirements

To install the Active Directory Agent, you need any one of the following:

- A Windows 2003 machine
- A Windows 2008 machine
- A Windows 2008 R2 machine

Note Windows 2003 R2 is not supported.

Note Internationalization is not supported.

This AD Agent machine could be one of the Active Directory domain controller machines that you will be monitoring, or it can be a separate, dedicated, Windows machine.

If your solution requires multiple AD Agent machines to be installed, remember that:

- There is no limit on the number of AD Agent machines that are not domain controller machines.
- For a given AD domain, you can directly install the AD Agent on only one domain controller machine.

In all cases, an AD Agent machine must meet the minimum hardware specifications listed in Table 2-1.

Table 2-1 Minimum Required Hardware Specifications for the AD Agent Machine

Component	Specification
CPU	Intel Xeon 2.66 GHz Q9400 (Quad Core)
System memory	4 GB of SDRAM
Hard disk space	500 GB

Connectivity Requirements

For the AD Agent to function properly, it must be able to communicate freely with all the client devices, Active Directory domain controller machines, and target syslog servers that are configured with it. If Windows Firewall (or any other comparable third-party firewall software) is running on the AD Agent machine, or on any of the Active Directory domain controller machines, then the firewall software on each of these endpoints must be configured with the necessary exceptions to allow this communication to flow freely.

This section uses the Windows Firewall as an example and details the exceptions that must be defined on any of the endpoints that might be running Windows Firewall:

- Windows Firewall Exceptions to be Configured on the AD Agent Machine, page 2-3
- Windows Firewall Exceptions to be Configured on Each Separate Active Directory Domain Controller Machine, page 2-4

For any other comparable third-party firewall software, refer to that vendor's documentation on how to configure the corresponding exceptions.

Windows Firewall Exceptions to be Configured on the AD Agent Machine

If Windows Firewall is enabled on the AD Agent machine, then:

a. You must explicitly define Windows Firewall exceptions for the following programs:

- C:\IBF\adObserver\ADObserver.exe
- C:\IBF\radiusServer\runtime\win32\bin.build\rt_daemon.exe

If the AD Agent machine is running Windows Server 2008 or Windows Server 2008 R2, then you can use the following Windows command lines (each written on a single line) to define these exceptions:

- **netsh advfirewall firewall add rule name="Cisco AD Agent (AD Observer)" dir=in action=allow program="C:\IBF\adObserver\ADObserver.exe" enable=yes**
- **netsh advfirewall firewall add rule name="Cisco AD Agent (RADIUS Server)" dir=in action=allow program="C:\IBF\radiusServer\runtime\win32\bin.build\rt_daemon.exe" enable=yes**

If the AD Agent machine is running Windows Server 2003 (with SP1 or later installed), then you can use the following Windows command lines (each written on a single line) to define these exceptions:

- **netsh firewall add allowedprogram C:\IBF\adObserver\ADObserver.exe "Cisco AD Agent (AD Observer)" ENABLE**
- **netsh firewall add allowedprogram C:\IBF\radiusServer\runtime\win32\bin.build\rt_daemon.exe "Cisco AD Agent (RADIUS Server)" ENABLE**

Note The original Windows Server 2003 did not support Windows Firewall. Windows Server 2003 SP1 added support for Windows Firewall, but left it disabled by default. Windows Server 2003 SP2 enables Windows Firewall by default.

1. If you use the **adacfg dc create** command to configure any Active Directory domain controller machine that is separate from the AD Agent machine, then you must also define a Windows Firewall exception on the AD Agent machine that will allow the necessary WMI-related communication.

Note In every case where the Windows Firewall is enabled on the AD Agent machine and on another separate, domain controller machine, and the AD Agent must communicate with the AD domain controller machine, each machine must have an exception for WMI. If the Windows Firewall is not running on either one of these machines, then a WMI exception is not required on that particular machine. A WMI exception is also not required when there is one single AD domain controller and the AD Agent is running on that same machine.

If the AD Agent machine is running Windows Server 2008 or Windows Server 2008 R2, then you can use the following Windows command line (written on a single line) to configure this WMI-related exception:

```
netsh advfirewall firewall set rule group="Windows Management Instrumentation (WMI)"
new enable=yes
```

If the AD Agent machine is running Windows Server 2003 (with SP1 or later installed), then you can use the following Windows command lines (each written on a single line) to configure this WMI-related exception:

- `netsh firewall add portopening protocol=tcp port=135 name="Cisco AD Agent (WMI_DCOM_TCP135)"`
- `netsh firewall add allowedprogram program=%windir%\system32\wbem\unsecapp.exe name="Cisco AD Agent (WMI_UNSECAPP)"`

Windows Firewall Exceptions to be Configured on Each Separate Active Directory Domain Controller Machine

For each separate Active Directory domain controller machine that is configured on the AD Agent machine using the **adacfg client create** command, if Windows Firewall is enabled on that separate domain controller machine, then you must define a Windows Firewall exception on that particular domain controller machine that will allow the necessary WMI-related communication.

If that domain controller machine is running Windows Server 2008 or Windows Server 2008 R2, then you can configure this WMI-related exception using the following Windows command line (written on a single line):

```
netsh advfirewall firewall set rule group="Windows Management Instrumentation (WMI)" new enable=yes
```

If that domain controller machine is running Windows Server 2003 (with SP1 or later installed), then you can configure this WMI-related exception using the following Windows command line (written on a single line):

```
netsh firewall set service RemoteAdmin enable
```

List of Open Ports

Table 2-2 lists some of the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) ports that the AD Agent uses for communication with client devices and Active Directory domain controllers. These ports must be open on the AD Agent.

Note This list does not include the dynamically allocated (random) port numbers that are used by WMI.

Table 2-2 List of Open Ports on the AD Agent

Port No.	Protocol	Service
8888 (on the local host)	TCP	Configuration changes
514	UDP	Syslog
1645 (on all interfaces)	UDP	Legacy RADIUS
1646 (on all interfaces)	UDP	Legacy RADIUS accounting

Table 2-2 List of Open Ports on the AD Agent (continued)

Port No.	Protocol	Service
1812 (on all interfaces)	UDP	RADIUS
1813 (on all interfaces)	UDP	RADIUS accounting

The port numbers for configuration changes and RADIUS are hardwired and not configurable. No other software application that uses these ports should be running on the AD Agent machine. For example, the AD Agent machine must not be running another RADIUS server.

Active Directory Requirements

For the AD Agent to communicate with the domain controllers, you must ensure that the following prerequisites are met:

- Each individual AD domain controller machine through which users will be authenticating during login and whose Security Log will be monitored by the AD Agent must run one of the following supported versions of Windows Server:
  - Windows Server 2003
  - Windows Server 2008
  - Windows Server 2008 R2

Note Windows Server 2003 R2 is not supported.

Note Internationalization is not supported

- Also, each individual domain controller machine running Windows Server 2008 or Windows Server 2008 R2 must have the appropriate Microsoft hotfixes installed. You must install the hotfixes regardless of whether the AD Agent is installed directly on the domain controller machine, or is monitoring the domain controller machine remotely.

  For domain controller machines running Windows Server 2008, the following two Microsoft hotfixes must be installed:

  a. http://support.microsoft.com/kb/958124

     This patch fixes a memory leak in Microsoft's WMI, which if left unfixed can prevent the AD Agent from successfully connecting with that domain controller and achieving an "up" status.

  b. http://support.microsoft.com/kb/973995

     This patch fixes a memory leak in Microsoft's WMI, which if left unfixed can sporadically prevent Active Directory from writing the necessary authentication-related events to the Security Log for that domain controller and would prevent the AD Agent from learning about the mappings corresponding to some of the user logins that authenticate through that domain controller.

  For domain controller machines running Windows Server 2008 R2, the following Microsoft hotfix must be installed (unless SP1 is installed):

http://support.microsoft.com/kb/981314

This patch fixes a memory leak in Microsoft's WMI, which if left unfixed can sporadically prevent Active Directory from writing the necessary authentication-related events to the Security Log for that domain controller and would prevent the AD Agent from learning about the mappings corresponding to some of the user logins that authenticate through that domain controller.

- Similarly, each individual AD domain controller machine through which users will be authenticating during login and whose Security Log will be monitored by the AD Agent must have an "Audit Policy" (part of the "Group Policy Management" settings) that allows successful logons to generate the necessary events in the Windows Security Log of that domain controller machine. See "Configuring AD Agent to Obtain Information from AD Domain Controllers" section on page 2-9.
- Before you configure even a single domain controller machine using the **adacfg dc create** command, ensure that the AD Agent machine is first joined to a domain (for example, domain *J*) that has a trust relationship with each and every domain (for example, domain *D[i]*) that it will monitor for user authentications (through the domain controller machines that you will be configuring on the AD Agent machine).

  Depending on your Active Directory domain structure, the following scenarios are possible:

  1. Single Forest, Single Domain—There is only one domain, *D[i]* for all domain controller machines, which is one and the same as domain *J*. The AD Agent machine must first be joined to this single domain, and since no other domains are involved, there is no need to configure any trust relationship with any other domain.
  2. Single Forest, Multiple Domains—All the domains in a single forest already have an inherent two-way trust relationship with each other. Thus, the AD Agent must first be joined to one of the domains, *J*, in this forest, with this domain *J* not necessarily being identical to any of the domains *D[i]* corresponding to the domain controller machines. Because of the inherent trust relationship between domain *J* and each of the domains *D[i]*, there is no need to explicitly configure any trust relationships.
  3. Multiple Forests, Multiple Domains—It is possible that domain *J* might belong to a forest that is different than the forest to which one or more of the domains *D[i]* corresponding to the domain controller machines belong. In this case, you must explicitly ensure that each of the domains *D[i]* has an effective trust relationship with domain *J*, in at least one of the following two ways:

     a. A two-way external trust relationship can be established between the two domains, *D[i]* and *J*
     b. A two-way forest trust relationship can be established between the the forest corresponding to domain *D[i]* and the forest corresponding to domain *J*

  To configure trust relationships, choose **Start > All Programs > Administrative Tools > Active Directory Domains and Trusts**.

Note If you ignore this requirement, and the AD Agent machine is not joined to a domain that has the necessary trust relationship with the domain associated with a particular DC machine, then your attempts to configure that DC machine, using the **adacfg dc create** command might appear to succeed. However, that DC machine might begin to have various problems, such as an extremely high CPU loading.

Installing Active Directory Agent

To install Active Directory Agent, complete the following steps:

Step 1 Copy the Active Directory Agent installer executable file to the Windows machine where you are going to install the Active Directory Agent.

Step 2 Run the **AD_Agent-v1.0.0.32-build-539.Installer.exe** file.

The Cisco AD Agent Setup dialog box appears.

Step 3 Click **Yes** to proceed with the installation.

The installer will install the AD Agent in the C:\IBF\ directory of your Windows machine. You can view the progress of the installation procedure. If the installation was successful, you will see a Completed message.

Step 4 Click **Close** to close the installer.

Confirming Active Directory Agent Installation

To confirm if the Active Directory Agent is running after installation, complete the following steps:

Step 1 Go to the Windows Command Line Prompt (**Start > All Programs > Accessories > Command Prompt**).

Step 2 Type **cd C:\IBF\CLI**.

Step 3 Type **adactrl.exe show running**.

You will see an output similar to the following:

```
running C:\\IBF\\watchdog\\radiusServer.bat since 2010-12-27 T15:32:31
running C:\\IBF\\watchdog\\adObserver.bat since 2010-12-27 T15:32:38
```

This output provides information on the date and time from which the AD Agent internal processes are running on this machine.

Uninstalling Active Directory Agent

To uninstall the Active Directory Agent, complete the following steps:

Step 1 Go to the C:\IBF\ folder.

The Active Directory Agent is by default installed in the C:\IBF\ directory of your Windows machine.

Step 2 Run the **AD_Agent.Uninstaller.exe** file.

The AD Agent is uninstalled.

Configuring Active Directory Agent

After you install the AD Agent, you must first ensure that if any firewall such as Windows Firewall is running on the AD Agent machine, then the necessary exceptions are configured on the AD Agent machine as described in "Windows Firewall Exceptions to be Configured on the AD Agent Machine" section on page 2-3. Then, you must configure the AD Agent with:

- Each individual Active Directory domain controller machine through which users will be authenticating during their logins, and whose Security Log will be monitored by the AD Agent to learn of new mappings through that particular domain controller.

Note You must also include any backup domain controller machines that you are deploying.

- Client devices, such as ASA devices, that have been properly configured to obtain the IP-to-user-identity mappings from the AD Agent machine.

You can also configure AD Agent to send logs to a syslog server.

Note If you configure the AD Agent with a syslog server first before you configure the AD domain controllers and client devices, then troubleshooting information will also be available in the syslog server in addition to the localStore. Having this troubleshooting information in the syslog server might be helpful in case you run into any issues during the setup process.

After you install the AD Agent, wait for a short while (about 30 seconds) for the AD Agent to properly initialize before you issue any of the adacfg commands.

- If you issue any of the adacfg commands when the AD Agent is not running, you will see the following message:

```
Error: HTTP request sending failed with error "Couldn't connect to server"! For
further syntax information, use adacfg help.
```

- If you issue any of the adacfg commands before the AD Agent has fully initialized, you will see the following message:

```
Caught exception: Module PipConfigurator not initialized!
```

This section contains the following topics:

Note This section only describes the configuration that you should perform on the AD Agent. For a solution to work properly, you must configure the AD Agent and AD domain controllers in the client devices as well. Refer to the *ASA End-User Documentation* for more information.

Configuring AD Agent to Send Logs to a Syslog Server

You can configure the AD Agent to send logs to a syslog server for administrative purposes and also for obtaining troubleshooting information.

To configure AD Agent to send logs to a syslog server, complete the following steps:

Step 1 Log into your AD Agent Windows machine.

Step 2 From the command line prompt, type cd **C:\IBF\CLI**.

Step 3 Enter the following command:

adacfg syslog create -name <*syslog-target-nickname*> **-ip** <*IP-address*> [**-facility** <*syslog-facility*>]

where

- *syslog-target-nickname* is a friendly name that you assign to the syslog server.
- *IP-address* is the IP address of the syslog server.
- *syslog-facility* values range from LOCAL0 through LOCAL7. The default is LOCAL6.

You will see the following message:

```
Reply: Command completed successfully.
```

Configuring AD Agent to Obtain Information from AD Domain Controllers

Each individual Active Directory domain controller machine through which users will be authenticating during their logins must be separately configured on the AD Agent, so that the AD Agent will be able to learn new IP-to-user-identity mappings from that particular domain controller by monitoring its Security Log.

Note You must include any backup domain controller machines that you are deploying.

To configure AD Agent to obtain information from a particular AD domain controller machine, complete the following steps:

Step 1 Ensure that the AD domain controller machine is running a supported version of the Windows Server operating system, as described in "Active Directory Requirements" section on page 2-5.

Step 2 Ensure that if the AD domain controller machine is running Windows Server 2008 or Windows Server 2008 R2, then the appropriate Microsoft hotfixes are installed on that machine, as described in "Active Directory Requirements" section on page 2-5. There should be no AD domain controller machine running Windows Server 2008 or 2008 R2 without the specified hotfixes.

Step 3 Ensure that if any firewall software, such as Windows Firewall, is enabled on the AD domain controller machine, then the necessary firewall exceptions are defined on the AD domain controller machine, as described in "Windows Firewall Exceptions to be Configured on Each Separate Active Directory Domain Controller Machine" section on page 2-4.

Step 4 Ensure that the domain associated with the domain controller machine has the proper trust relationship with the domain to which the AD Agent machine is joined.

Step 5 Ensure that the “Audit Policy” (part of the "Group Policy Management" settings) allows successful logons to generate the necessary events in the Windows Security Log of that AD domain controller machine (this is normally the Windows default setting, but you must explicitly ensure that this setting is correct). To do this, choose **Start > Programs > Administrative Tools > Group Policy Management**. From the navigation pane on the left of Group Policy Management:

a. Navigate under **Domains** to the relevant domain(s).

b. Expand the navigation tree.

c. Right-click **Default Domain Policy**.

d. Choose the **Edit** menu item, which will bring up the Group Policy Management Editor.

e. From the navigation pane on the left of Group Policy Management Editor:

f. Choose **Default Domain Policy > Computer Configuration > Policies > Windows Settings > Security Settings**.

- For Windows Server 2003 or Windows Server 2008 (non-R2), choose **Local Policies > Audit Policy**. For the two Policy items, **Audit Account Logon Events** and **Audit Logon Events**, ensure that the corresponding **Policy Setting** for each of these either directly or indirectly includes the **Success** condition. To include the **Success** condition indirectly, the Policy Setting must be set to **Not Defined**, indicating that the effective value will be inherited from a higher level domain, and the Policy Setting for that higher level domain must be configured to explicitly include the **Success** condition.
- For Windows Server 2008 R2, choose **Advanced Audit Policy Configuration > Audit Policies > Account Logon**. For the two Policy items, **Audit Kerberos Authentication Service** and **Audit Kerberos Service Ticket Operations**, ensure that the corresponding **Policy Setting** for each of these either directly or indirectly includes the **Success** condition as described above.

g. If any **Audit Policy** item settings have been changed, you should then run “**gpupdate /force**” to force the new settings to take effect.

Step 6 Log into your AD Agent Windows machine.

Step 7 From the command line prompt, type cd **C:\IBF\CLI**.

Step 8 Enter the following command:

adacfg dc create -name <*DC-nickname*> **-host** <*DC-hostname-or-FQDN*> **-domain** <*full-DNS-name-of-AD-domain*> **-user** <*username-member-of-Domain-Admins-group*> **-password** <*password-of-user*>

where

- *DC-nickname* is a friendly name that you assign to the domain controller.
- *DC-hostname-or-FQDN* is the hostname or the fully qualified domain name of the AD domain controller machine to be monitored by the AD Agent.
- *full-DNS-name-of-AD-domain* is the full DNS name of the AD domain.
- *username-member-of-Domain-Admins-group* is the username of an existing account through which the security log of the domain controller machine will be monitored.

  This account must have the necessary privileges for reading the Security Log of the domain controller machine. You can easily and conveniently ensure this by specifying an account that belongs to the “Domain Admins” AD group for the domain specified with the “-domain” option.

  Alternatively, it is possible for nonmembers of the “Domain Admins” group to have the necessary privileges by satisfying all of the following requirements:

  - The account must belong to the “Distributed COM Users” AD group.

  - The account must have permission to access WMI namespaces (in particular, the "CIMV2" namespace) on the domain controller machine. You can configure this permission using the 'wmimgmt.msc' snap-in, or through Group Policy (to affect all domain controller machines). See http://blogs.msdn.com/b/spatdsg/archive/2007/11/21/set-wmi-namespace-security-via-gpo-script.aspx for more information.
  - The account must have permission to read the Security Event Log on the domain controller machine. You can configure this permission by setting the CustomSD key in the registry, or through the Group Policy (to affect all domain controller machines). See http://msdn.microsoft.com/en-us/library/aa363648%28v=vs.85%29.aspx for more information.
- *password-of-user* is the password corresponding to the username specified above.

You will see the following message:

```
Reply: Command completed successfully.
```

You can use the **adacfg dc list** command to view a list of the currently configured AD domain controller machines and their up or down status. You can periodically reenter this command to recheck the status of the AD domain controller machines.

After you run the **adacfg dc create** command for a particular AD domain controller, you must wait a short while (for about a minute or so), until the status of that AD domain controller changes from its initial "down" state to "up" or "down(no-retry)."

- The "up" state indicates that connectivity with that AD domain controller has been established. You might need to wait several more minutes (or even longer) from the time that a particular AD domain controller machine has reached the "up" state for the first time ever to the time that any historical mappings are retrieved from that machine and become visible through the **adacfg cache list** command.
- The "down(no-retry)" state indicates that connectivity could not be established (for example, due to incorrect credentials) and that the AD Agent will not re-attempt to establish connectivity.
- On the other hand, the "down" state indicates that currently the AD Agent does not have connectivity with that particular AD domain controller machine, but it will periodically re-attempt to establish connectivity.

You can also use the **adacfg dc erase** command to remove any domain controller configuration from the AD Agent.

See "adacfg dc list" section on page A-8, "adacfg cache list" section on page A-9, and "adacfg dc erase" section on page A-8 for more information on these commands.

Configuring AD Agent to Allow Client Devices to Obtain Information from AD Agent

You must configure the AD Agent with each individual client device (such as ASA) for the AD Agent to respond to requests from that particular client device to receive mapping information from this AD Agent.

Note A single AD Agent can support a maximum of 100 client devices (such as ASA devices).

To configure AD Agent to communicate with a particular client device, complete the following steps:

Step 1 Log into your AD Agent Windows machine.

Step 2 From the command line prompt, type cd **C:\IBF\CLI**.

Step 3 Enter the following command:

adacfg client create -name <*client-nickname*> **-ip** <*IP-address*>*[/*<*prefix-length-for-IP-range*>*]* **-secret** <*RADIUS-shared-secret*>

where

- *client-nickname* is a friendly name that you assign to the particular client device.
- *IP-address/*<*prefix-length-for-IP-range*> refers to the IP address of the particular client device and you can optionally define a subnet range.
- *RADIUS-shared-secret* is the shared secret that the RADIUS protocol uses for communication. This *secret* is the key that is configured on the particular client device.

Note Ensure that you enter the correct RADIUS-shared-secret. Otherwise, requests from that particular client device will be ignored.

You will see the following message:

```
Reply: Command completed successfully!
```

You can use the **adacfg client list** command to view a list of currently configured client devices and the **adacfg client erase** command to remove any client device configuration from the AD Agent. See "adacfg client list" section on page A-5 and "adacfg client erase" section on page A-5 for more information on these commands.

Step 4 Follow the instructions provided with the particular client device to configure the client device to recognize this AD Agent machine.

APPENDIX A

Active Directory Agent Command Reference

This appendix contains an alphabetical listing of commands specific to the Active Directory Agent. The commands comprise these modes:

- adactrl—Used to start, stop, and restart the AD Agent, and to monitor its running status.
- adacfg—Used to configure the Active Directory Agent with client devices, Active Directory domain controllers, and Syslog servers.

Each of the commands in this appendix is followed by a brief description of its use, command syntax, usage guidelines, and an example.

This appendix contains the following sections:

AD Agent Control Commands

This section describes the following commands:

- **adactrl help**
- **adactrl restart**
- **adactrl show running**
- **adactrl start**
- **adactrl stop**
- **adactrl version**

Note All the adactrl commands are case-sensitive.

adactrl help

To view a list of adactrl commands and their syntax.

Syntax

adactrl help

Example

```
C:\IBF\CLI>adactrl help
Cisco AD Agent adctrl -- version 1.0.0.32, build 539
    Usage: adactrl COMMAND
    where COMMAND can be:
        start        - to start the AD Agent
        stop         - to stop the AD Agent
        restart      - to restart the AD Agent
        show running - to show the running status of the AD Agent
        version      - to view info on AD Agent version currently installed
        help         - to view this help
```

adactrl restart

To stop and restart the AD Agent.

Syntax

adactrl restart

Example

```
C:\IBF\CLI>adactrl restart
OK
```

adactrl show running

To view the status of the AD Agent's internal components: radiusServer and adObserver.

Syntax

adactrl show running

Example

```
C:\IBF\CLI>adactrl show running
running C:\\IBF\\watchdog\\radiusServer.bat since 2011- 1- 5 T10:25:44
running C:\\IBF\\watchdog\\adObserver.bat since 2011- 1- 5 T10:25:44
```

adactrl start

To start the AD Agent.

Syntax

adactrl start

Example

```
C:\IBF\CLI>adactrl start
OK
```

adactrl stop

To stop the AD Agent.

Syntax

adactrl stop

Example

```
C:\IBF\CLI>adactrl stop
OK
```

adactrl version

To view the version of AD Agent that is installed on your Windows machine.

Syntax

adactrl version

Example

```
C:\IBF\CLI>adactrl version
Cisco AD Agent adactrl -- version 1.0.0.32, build 539
(Built from sources last modified 2011-04-21 12:20:17 +0300)
```

AD Agent Configuration Commands

This section describes the following commands:

- **adacfg help**
- **adacfg help client**
- **adacfg client create**
- **adacfg client erase**
- **adacfg client list**
- **adacfg client status**
- **adacfg help dc**
- **adacfg dc create**
- **adacfg dc erase**
- **adacfg dc list**
- **adacfg help cache**
- **adacfg cache list**
- **adacfg cache clear**
- **adacfg help options**
- **adacfg options list**
- **adacfg options set**

- adacfg help syslog
- adacfg syslog create
- adacfg syslog erase
- adacfg syslog list
- adacfg version

Note The adacfg commands are not case-sensitive.

adacfg help

To view the top-level summary of the **adacfg** command syntax.

Syntax

adacfg help

Example

```
C:\IBF\CLI>adacfg help
Cisco AD Agent adacfg -- version 1.0.0.32, build 539
    Usage: adacfg [COMMAND]
    where COMMAND can be:
        client       - to manage client-devices of AD Agent
        dc           - to manage AD domain-controller machines monitored by AD Agent
        syslog       - to manage syslog-targets of AD Agent
        options      - to manage configurable settings for AD Agent
        cache        - to manage cache of identity-mappings maintained by AD Agent
        version      - to view info on AD Agent version currently installed
        help         - to view this help
        help COMMAND - to view the help for specified COMMAND
```

adacfg help client

To view a detailed syntax summary of the client-related **adacfg** commands.

Syntax

adacfg help client

Example

```
C:\IBF\CLI>adacfg help client
Cisco AD Agent adacfg -- version 1.0.0.32, build 539
    Usage:  adacfg client [SUBCOMMAND] [ARGS]
    where SUBCOMMAND can be:
            create  - to configure a new client
            list    - to list all previously configured clients
            erase   - to erase a previously configured client
            status  - to view status of clients subscribed for notification
            help    - to view this help
    detailed syntax (write command on a single line!):
            adacfg client create -name <client-nickname>
                                 -ip <IP-address>[/<prefix-length-for-IP-range>]
                                 -secret <RADIUS-shared-secret>
```

```
adacfg client list
adacfg client erase -name <client-nickname>
adacfg client status
```

adacfg client create

To configure a new client device.

Syntax

adacfg client create -name <*client-nickname*> **-ip** <*IP-address*>*[/<prefix-length-for-IP-range>]* **-secret** <*RADIUS-shared-secret*>

where

- *client-nickname*—Any friendly name that you can assign to the client device.
- *IP-address*—The IP address of the client device.
- *prefix-length-for-IP-range*—You can optionally define an IP subnet range.
- *RADIUS-shared-secret*—The shared secret that the RADIUS protocol uses to communicate with the client device. This *secret* is the key that is configured on the client device.

Example

```
C:\IBF\CLI>adacfg client create -name asa1 -ip 10.77.202.1/32 -secret cisco123
Reply: Command completed successfully!
```

adacfg client erase

To erase a previously configured client.

Syntax

adacfg client erase -name <*client-nickname*>

where *client-nickname* is the name of the client device.

Example

```
C:\IBF\CLI>adacfg client erase -name asa1
Reply: Command completed successfully!
```

adacfg client list

To list all previously configured client devices.

Syntax

adacfg client list

Example

```
C:\IBF\CLI>adacfg client list
Name  IP/Range
----  ---------
asa1  10.77.204.2
asa2  10.77.101.3
asa3  10.77.101.4
```

adacfg client status

To view the sync status of the clients that are subscribed for notification (on-demand queries that also include a request for notification, or requests to register for replication).

Syntax

adacfg client status

Example

```
C:\IBF\CLI>adacfg client status
Subscribed-IP  Sync Status
-------------  ------------
10.77.101.2    In-Sync
10.77.101.3    Out-Of-Sync
10.77.101.4    Out-Of-Sync
10.77.101.5    In-Sync
```

adacfg help dc

To view a detailed syntax summary of the DC-related **adacfg** commands.

Syntax

adacfg help dc

Example

```
C:\IBF\CLI>adacfg help dc
Cisco AD Agent adacfg -- version 1.0.0.32, build 539
    Usage:  adacfg dc [SUBCOMMAND] [ARGS]
    where SUBCOMMAND can be:
            create  - to configure a new AD domain-controller machine
            list    - to list all previously configured AD domain-controller machines
            erase   - to erase a previously configured AD domain-controller machine
            help    - to view this help
    detailed syntax (write command on a single line!):
            adacfg dc create -name <DC-nickname>
                             -host <DC-hostname-or-FQDN>
                             -domain <full-DNS-name-of-AD-domain>
                             -user <username-member-of-Domain-Admins-group>
                             -password <password-of-user>
            adacfg dc list
            adacfg dc erase -name <DC-nickname>
```

adacfg dc create

To configure a new AD domain controller machine.

Syntax

adacfg dc create -name <*DC-nickname*> **-host** <*DC-hostname-or-FQDN*> **-domain** <*full-DNS-name-of-AD-domain*> **-user** <*username-member-of-Domain-Admins-group*> **-password** <*password-of-user*>

where

- *DC-nickname*—Name of the Active Directory domain controller.
- *DC-hostname-or-FQDN*—The hostname of the AD domain controller or the fully qualified domain name (FQDN) of the Active Directory domain controller.
- *full-DNS-name-of-AD-domain*—The full DNS name of the AD domain.
- *username-member-of-Domain-Admins-group*—The username of an existing account through which the security log of the domain controller machine will be monitored.

  This account must have the necessary privileges for reading the Security Log of the domain controller machine. You can easily and conveniently ensure this by specifying an account that belongs to the “Domain Admins” AD group for the domain specified with the “-domain” option.

  Alternatively, it is possible for nonmembers of the “Domain Admins” group to have the necessary privileges by satisfying all of the following requirements:

  – The account must belong to the “Distributed COM Users” AD group.

  – The account must have permission to access WMI namespaces (in particular, the “CIMV2” namespace) on the domain controller machine. You can configure this permission using the ‘wmimgmt.msc’ snap-in, or through Group Policy (to affect all domain controller machines). See http://blogs.msdn.com/b/spatdsg/archive/2007/11/21/set-wmi-namespace-security-via-gpo-script.aspx for more information.

  – The account must have permission to read the Security Event Log on the domain controller machine. You can configure this permission by setting the CustomSD key in the registry, or through the Group Policy (to affect all domain controller machines). See http://msdn.microsoft.com/en-us/library/aa363648%28v=vs.85%29.aspx for more information.
- *password-of-user*—The password corresponding to the username specified above.

Example

```
C:\IBF\CLI>adacfg dc create -name abc-dc1 -host amer.acs.com -domain acs.com -user xyz
-password axbycz
Warning: please make sure that this DC machine has:
  [1] all necessary patches installed, and
  [2] a properly configured Audit Policy.
  For more details, visit:
  http://www.cisco.com/en/US/docs/security/asa/asa84/release/notes/README_FIRST.html

Command completed successfully!
```

adacfg dc erase

To erase a previously configured AD domain controller machine.

Syntax

adacfg dc erase -name <*DC-nickname*>

Example

```
C:\IBF\CLI>adacfg dc erase -name abc-dc1
Reply: Command completed successfully!
```

adacfg dc list

To list all previously configured AD domain controller machines.

Syntax

adacfg dc list

Example

```
C:\IBF\CLI>adacfg dc list
C:\IBF\CLI>adacfg dc list
Name     Host/IP       Username     Domain-Name   Latest Status
----     -------       --------     -----------   -------------
abc-dc1  amer.acs.com  domainAdmin  ACS           up
abc-dc2  amer2.acs.com domainAdmin                down
abc-dc3  amer3.acs.com domainAdmin                down(no-retry)
```

adacfg help cache

To view a detailed syntax summary of the cache-related **adacfg** commands.

Syntax

adacfg help cache

Example

```
C:\IBF\CLI>adacfg help cache
Cisco AD Agent adacfg -- version 1.0.0.32, build 539
    Usage:  adacfg cache [SUBCOMMAND]
    where SUBCOMMAND can be:
            list  - to view the currently cached mappings
            clear - to clear the currently cached mappings
            help  - to view this help
    detailed syntax:
            adacfg cache list
            adacfg cache clear
```

adacfg cache list

To view the currently cached mappings.

Syntax

adacfg cache list

Example

```
C:\IBF\CLI>adacfg cache list
IP           User-Name Domain Response-to-Probe Mapping-Type Mapping-Origin Create-Time
--           --------- ------ ----------------- ------------ -------------- -----------
10.77.100.1  User1     AD1    true              DC           AD1  2011-01-05T09:37:17Z
10.77.100.2  User2     AD1    true              DC           AD1  2011-01-05T09:37:21Z
```

adacfg cache clear

To clear the currently cached mappings.

Syntax

adacfg cache clear

Example

```
C:\IBF\CLI>adacfg cache clear
Removed 10 records.
```

Note You must allow some time for the cache to be completely cleared internally, depending on the number of mappings that are currently cached.

A very large number of mappings in the cache can take a minute or so to be completely cleared. Meanwhile, interim invocations of the **adacfg cache list** command might appear to show that mappings still exist, or even return an SQL error that states the "database is locked," but you can safely ignore these results. Once the clear cache operation ultimately completes internally, the **adacfg cache list** command will return a "Total mappings count" of 0.

adacfg help options

To view a detailed syntax summary of the options-related **adacfg** commands.

Syntax

adacfg help options

Example

```
C:\IBF\CLI>adacfg help options
Cisco AD Agent adacfg -- version 1.0.0.32, build 539
    Usage:  adacfg options [SUBCOMMAND] [ARGS]
    where SUBCOMMAND can be:
            list - to view the current settings of the configurable options
            set  - to configure one or more of the configurable options
```

```
        help - to view this help
detailed syntax:
        adacfg options list
        adacfg options set [-<optionName> <optionValue>] [...]

      an <optionName>/<optionValue> pair can be:

        [-userLogonTTL <number-of-minutes>]
          Time duration after which logged-in user is marked as being logged-out.

        [-dcStatusTime <number-of-seconds>]
          Time span between consecutive monitorings of DC-machine up/down status.

        [-dcHistoryTime <number-of-seconds>]
          Amount of time before the present from which to start reading
          the security logs of DC-machines that are configured
          (via 'adacfg dc create') for the first time ever.

        [-notifyAttributes <text>]
          Comma-separated list of attributes to be sent in notifications to
          subscribed client-devices.

          Fully expanded list:
            domain,time-stamp,responds-to-probe,mapping-type,mapping-origin

          Wildcard equivalent to the fully expanded list:
            *

        [-logLevel <level>]
          Logging level for the customer logs (localStore and syslogs).

          Valid values: FATAL, ERROR, WARN, INFO, or DEBUG

          Default value: INFO
```

adacfg options list

To view the current settings of the configurable options.

Syntax

adacfg options list

Example

```
C:\IBF\CLI>adacfg options list
Option           Value
---------------  -----
userLogonTTL     1440
dcHistoryTime    86400
dcStatusTime     60
notifyAttributes *
logLevel         INFO
```

adacfg options set

To configure one or more of the configurable options.

Syntax

adacfg options set [-<optionName> <optionValue>] [...]

where optionName and optionValue pairs could be any or all of the following:

- **[userLogonTTL** <*number-of-minutes*>**]**—Time duration after which logged-in user is marked as being logged-out.
- **[dcStatusTime** <*number-of-seconds*>**]**—Time span between consecutive monitorings of DC-machine up/down status.
- **[dcHistoryTime** <*number-of-seconds*>**]**—Amount of time before the present from which to start reading the security logs of DC-machines that are configured (via 'adacfg dc create') for the first time ever.
- **[notifyAttributes** <*text*>**]**—Comma-separated list of attributes to be sent in notifications to subscribed client-devices, which could be any or all of the following attributes:
  - **domain, time-stamp, responds-to-probe, mapping-type, mapping-origin**
  - ***** (wildcard equivalent of all the attributes)
- **[logLevel** <*level*>**]**—Logging level for the customer logs (localStore and syslogs). Valid values include FATAL, ERROR, WARN, INFO, and DEBUG. The default level is INFO.

Note The AD Agent generates some of its Customer Log messages using the “NOTICE” logging level (which falls between the “INFO” and “WARN” levels), but you cannot explicitly choose “NOTICE” as a setting for ‘logLevel’ using the **adacfg options set -logLevel** command. See Appendix B, “Customer Log Messages,” for more details.

adacfg help syslog

To view a detailed syntax summary of the syslog-related **adacfg** commands.

Syntax

adacfg help syslog

Example

```
C:\IBF\CLI>adacfg help syslog

Cisco AD Agent adacfg -- version 1.0.0.32, build 539
    Usage:  adacfg syslog [SUBCOMMAND] [ARGS]
    where SUBCOMMAND can be:
            create  - to configure a new syslog-target
            list    - to list all previously configured syslog-targets
            erase   - to erase a previously configured syslog-target
            help    - to view this help
    detailed syntax (write command on a single line!):
            adacfg syslog create -name <syslog-target-nickname>
                                 -ip <IP-address>
                                [-facility <syslog-facility>]
              valid syslog facility values:  LOCAL0 - LOCAL7
              default syslog facility value: LOCAL6
```

```
adacfg syslog list
adacfg syslog erase -name <syslog-target-nickname>
```

adacfg syslog create

To configure a new syslog target.

Syntax

adacfg syslog create -name <*syslog-target-nickname*> **-ip** <*IP-address*> [**-facility** <*syslog-facility*>]

where

- *syslog-target-nickname*—Name of the syslog server.
- *IP-address*—IP address of the syslog server.
- *syslog-facility*—Facility values range from LOCAL0 to LOCAL7. The default is LOCAL6.

Example

```
C:\IBF\CLI>adacfg syslog create -name mysyslog -ip 10.77.202.1 -facility LOCAL6
Reply: Command completed successfully!
```

adacfg syslog erase

To erase a previously configured syslog target.

Syntax

adacfg syslog erase -name <*syslog-target-nickname*>

where *syslog-target-nickname* is the name of the syslog target connected to the AD Agent.

Example

```
C:\IBF\CLI>adacfg syslog erase -name mysyslog
Reply: Command completed successfully.
```

adacfg syslog list

To list all the previously configured syslog targets.

Syntax

adacfg syslog list

Example

```
C:\IBF\CLI>adacfg syslog list
Name      IP           Facility
-----     ---          ---------
mysyslog  10.77.202.4  LOCAL6
```

adacfg version

To view the version of the AD Agent installed on your Windows machine.

Syntax

adacfg version

Example

```
C:\IBF\CLI>adacfg version
Cisco AD Agent adacfg -- version 1.0.0.32, build 539
<Built from sources last modified 2011-04-21 12:20:17 +0300>
```

APPENDIX **B**

Customer Log Messages

This appendix summarizes the various Customer Log messages that the AD Agent can generate (based on the current value of the “logLevel” configurable option) under various functional categories.

You can use the **adacfg options set –logLevel** command to change the “logLevel” setting.

Note Some of these log messages are associated with the “NOTICE” logging level (which falls between the “INFO” and “WARN” levels), but “NOTICE” itself cannot be chosen as a setting for 'logLevel' using the above adacfg command.

The following scale shows the full spectrum of logging levels, in order of increasing verbosity, with “NOTICE” highlighted in Italics font to show that it is not a configurable option for “logLevel,” and “INFO” highlighted in bold face font to show that it is the default setting for “logLevel:”

FATAL < ERROR < WARN < *NOTICE* < **INFO** < DEBUG

Note When you troubleshoot problems, it is often helpful to raise the setting of “logLevel” from its default setting of “INFO” to the most verbose level “DEBUG.” However, this setting can have a negative impact on the performance of the AD Agent. We recommend that you restore the setting of “logLevel” after you have resolved your problems.

Similarly, lowering the “logLevel” from its default setting of “INFO” to a lesser verbose level “WARN” can have a positive impact on the performance of the AD Agent. However, this setting will prevent the output of any “INFO” or “NOTICE” level messages (that can potentially be important for administrative or auditing purposes).

A local archive of historical Customer Log messages is maintained inside the “C:\IBF\radiusServer\runtime\logs\localStore” directory. These log messages are also forwarded to any remote syslog targets that have been configured using the **adacfg syslog create** command.

Table B-1 lists the AD-Agent-specific messages that can be logged to the Customer Logs, with the message code, logging level, message class, message text, and a general description corresponding to each. This is not the full list of all messages that can be generated, and it does not include, for example, generic RADIUS-related messages or other miscellaneous messages.

Table B-1 AD Agent Log Messages

Message Code	Logging Level	Message Class	Message Text	Description
AD Agent Start/Stop				
(a more comprehensive set of messages is logged in the Windows Application Event Log. See Appendix C, "Windows Application Event Logs," for more information.				
31502	INFO	STARTUP-SHUTDOWN	Started Runtime	The "RADIUS Server" subcomponent of the AD Agent has been started.
Configuration Changes				
68000	NOTICE	IBF_CONFIG_CHANGE	Created DC configuration	A domain controller machine has been configured in the AD Agent using the **adacfg dc create** command.
68001	NOTICE	IBF_CONFIG_CHANGE	Deleted DC configuration	A domain controller machine configuration has been removed from the AD Agent using the **adacfg dc erase** command.
68002	NOTICE	IBF_CONFIG_CHANGE	Created RADIUS-client configuration	A client device has been configured in the AD Agent using the **adacfg client create** command.
68003	NOTICE	IBF_CONFIG_CHANGE	Deleted RADIUS-client configuration	A client device configuration has been removed from the AD Agent using the **adacfg client erase** command.
Mapping Updates				
12862	INFO	IBF_RADIUS_SERVER	Updated mapping in Identity Cache	An IP-address-to-user-identity mapping has been added or updated in the internal cache of the AD Agent.
12855	INFO	IBF_RADIUS_SERVER	Dropped Identity Cache mapping-update due to userLogonTTL	An incoming IP-address-to-user-identity mapping update has been ignored by the AD Agent because the logon time occurred too far in the past (relative to the "userLogonTTL" configurable option).
12856	INFO	IBF_RADIUS_SERVER	Dropped Identity Cache mapping-update: older than existing mapping	An incoming IP-address-to-user-identity mapping update has been ignored by the AD Agent because the associated logon time was earlier that that of the one currently cached by the AD Agent for the same IP address.

Table B-1 AD Agent Log Messages (continued)

Message Code	Logging Level	Message Class	Message Text	Description
12859	INFO	IBF_RADIUS_SERVER	Dropped Identity Cache mapping-update having timestamp in future	An incoming IP-address-to-user-identity mapping update has been ignored by the AD Agent because the associated logon time is in the future.
12861	INFO	IBF_RADIUS_SERVER	Dropped Identity Cache mapping-update: same time as existing mapping	An incoming IP-address-to-user-identity mapping update has been ignored by the AD Agent because the associated logon time is identical to the one currently cached by the AD Agent for the same IP address and username.
12867	WARN	IBF_RADIUS_SERVER	Approaching stress limit on Identity Cache mapping-updates	AD Agent is approaching its maximum capacity limit on the number of cached mappings. Currently, over 100,000 mappings are cached. If the cache occupancy reaches 200,000 mappings, the AD Agent starts to ignore subsequent incoming mapping updates.
12868	ERROR	IBF_RADIUS_SERVER	Dropped Identity Cache mapping-updates: stress limit exceeded	The AD Agent has reached its maximum capacity limit of 200,000 mappings, and is now ignoring all new incoming mapping updates.
12893	INFO	IBF_RADIUS_SERVER	Deleted mapping in Identity Cache	An IP-address-to-user-identity mapping has been removed from the internal cache of the AD Agent.
Synch Requests				
12869	INFO	IBF_RADIUS_SERVER	Detected Synch request with registration for notifications	A client device has requested to receive a session data snapshot of all mappings currently found in the cache of the AD Agent. The client requests to be registered with the AD Agent for replication.
12860	INFO	IBF_RADIUS_SERVER	Detected Synch request with no registration for notifications	A client device has requested to receive a session data snapshot of all mappings currently found in the cache of the AD Agent. The client does not want to be registered with the AD Agent for replication.

Table B-1 AD Agent Log Messages (continued)

Message Code	Logging Level	Message Class	Message Text	Description
12870	INFO	IBF_RADIUS_SERVER	Detected Synch request without change to registration state	A client device has requested to receive a session data snapshot of all mappings currently found in the cache of the AD Agent. The client does not want to change its replication-related state with the AD Agent.
12871	INFO	IBF_RADIUS_SERVER	Detected deregistration Request	A client device has requested not be registered for replication with the AD Agent.
12872	WARN	IBF_RADIUS_SERVER	Approaching capacity limit on max registrations	The AD Agent is approaching its maximum limit on the number of unique client devices that can be simultaneously registered for notification. Currently, over 100 unique client devices can be registered for notification. If this number reaches 120 unique client devices, the AD Agent starts to ignore subsequent incoming requests to register for notification.
12873	ERROR	IBF_RADIUS_SERVER	Dropped registrations: capacity limit exceeded	The AD Agent has reached its maximum limit of 120 unique client devices that can be simultaneously registered for notification, and is now ignoring all new incoming requests to register for notification.
CoA-Based Traffic				
12884	INFO	IBF_RADIUS_SERVER	Sent RADIUS CoA-Request with Notification to PEP	The AD Agent has proactively notified a client device about a mapping that has changed since the time it was provided to that client device. This notification update to the client device is sent using a RADIUS CoA-Request packet.
11223	INFO	Dynamic-Authorization	Received CoA ACK response	The AD Agent has received a RADIUS CoA-ACK packet sent by a client device. The client device acknowledges receipt of the CoA-Request packet sent by the AD Agent.
11224	INFO	Dynamic-Authorization	Received CoA NAK response	The AD Agent has received a RADIUS CoA-NAK packet sent by a client device. The client device acknowledges a problem about a CoA-Request packet sent by the AD Agent.

Table B-1 AD Agent Log Messages (continued)

Message Code	Logging Level	Message Class	Message Text	Description
Session Data Snapshot Transfers				
12878	INFO	IBF_RADIUS_SERVER	Stopping current transfer of session data snapshot	The currently in-progress transfer of a session data snapshot to a client device is being stopped.
12881	INFO	IBF_RADIUS_SERVER	Started transfer of session data snapshot	A new transfer of a session data snapshot to a client device is started. **Note** This log item will be used only for snapshot transfers involving two or more RADIUS packets. The first such packet will be marked with #12881. The last such packet will be marked with #12883. Any packets in between will be marked with #12882.
12882	INFO	IBF_RADIUS_SERVER	Continued transfer of session data snapshot	The currently in-progress transfer of a session data snapshot to a client device is continued. **Note** This log item will be used only for large snapshot transfers involving three or more RADIUS packets. The first such packet will be marked with #12881, and the last such packet will be marked with #12883. All of the packets in between will be marked with #12882.
12883	INFO	IBF_RADIUS_SERVER	Finished transfer of session data snapshot	The transfer of a session data snapshot to a client device has successfully been completed. Very small snapshot transfers, which can fit into a single RADIUS packet will be marked only with this log item (after transfer completion), but not with any #12881 or #12882 log items.
On-Demand Queries				
12864	INFO	IBF_RADIUS_SERVER	Detected On-Demand Entity-Request from PEP	A client device has requested to receive a mapping for a particular IP address (either with or without a request for subsequent notification).
12866	INFO	IBF_RADIUS_SERVER	Could not find identity in Identity Cache	The AD Agent could not find in its cache a mapping having the requested IP address.

Table B-1 AD Agent Log Messages (continued)

Message Code	Logging Level	Message Class	Message Text	Description
Keepalive Requests				
12885	INFO	IBF_RADIUS_SERVER	Detected Keepalive Request from PEP	A client device has sent a keepalive request to the AD Agent.
Domain Status Queries				
12890	INFO	IBF_RADIUS_SERVER	Prepared Domain Status Query-Response	The AD Agent has prepared, and is about to send, a response to a client device that has requested the AD Agent's connectivity status with regard to a specific Active Directory domain.
Domain Controller Status Tracking				
12892	INFO	IBF_AD_MONITOR	ActiveDirectory domain controller status changed	The AD Agent has detected a change in the up/down status of a domain controller machine.
Miscellaneous				
12888	WARN	IBF_RADIUS_SERVER	Internal Warning	The AD Agent has experienced an internal problem.
12889	ERROR	IBF_RADIUS_SERVER	Internal Error	The AD Agent has experienced an internal problem.
General Pass/Fail Status (each such log entry refers to corresponding entries having an identical value for their associated 'IbfSessionID' attribute.)				
5200	NOTICE	Passed-Attempt	IBF request succeeded	AD Agent has been able to successfully process a request (a Synch Request, an On-Demand Query, a Keepalive Request, or a Domain Status Query) previously received from a client device.
5400	NOTICE	Failed-Attempt	IBF request failed	AD Agent has been unable to successfully process a request (a Synch Request, an On-Demand Query, a Keepalive Request, or a Domain Status Query) previously received from a client device, due to a reason encoded in the value of the 'FailureReason' attribute. In this or any other similar (but generic) 'Failed-Attempt' log entry, possible 'FailureReason' codes are presented in the next subsection of this table.

Table B-1 AD Agent Log Messages (continued)

Message Code	Logging Level	Message Class	Message Text	Description
5405	NOTICE	Failed-Attempt	RADIUS Request dropped	AD Agent has received a RADIUS packet, but is silently dropping it, due to a reason encoded in the value of the 'FailureReason' attribute. In this or any other similar (but generic) 'Failed-Attempt' log entry, possible 'FailureReason' codes are presented in the next subsection of this table.
5413	NOTICE	Failed-Attempt	RADIUS Accounting-Request dropped	AD Agent has received a RADIUS Accounting-Request packet, but is silently dropping it, due to a reason encoded in the value of the 'FailureReason' attribute. In this or any other similar (but generic) 'Failed-Attempt' log entry, possible 'FailureReason' codes are presented in the next subsection of this table.
Potential 'FailureReason' Messages				
(any 'FailureReason' code specified in a 'Failed-Attempt' log entry but not found in this subsection of the table should be regarded as an internal AD Agent error.)				
11007	DEBUG	RADIUS	Could not locate Network Device or AAA Client	A RADIUS packet has been received from an IP address not associated with any currently-configured client devices. Make sure that this device is configured, using 'adacfg client create'.
11011	WARN	RADIUS	RADIUS listener failed	Could not open one or more of the UDP ports used for receiving RADIUS requests. Make sure that no other processes on the AD Agent machine are using ports 1812, 1813, 1645, or 1646.
11012	ERROR	RADIUS	RADIUS packet contains invalid header	A RADIUS packet having an invalid header has been received. Make sure that the client device is compatible with AD Agent and is functioning properly.
11013	INFO	RADIUS	RADIUS packet already in the process	An incoming RADIUS request has been ignored by AD Agent, because it is a duplicate of another previously received packet that is currently being processed.
11014	ERROR	RADIUS	RADIUS packet contains invalid attribute(s)	A RADIUS packet having invalid attributes has been received. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly.

Table B-1 AD Agent Log Messages (continued)

Message Code	Logging Level	Message Class	Message Text	Description
11021	ERROR	RADIUS	RADIUS could not decipher password. packet missing necessary attributes	A RADIUS packet having a password that could not be deciphered has been received. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly. Make sure that the same RADIUS shared secret has been properly configured, both in the client device and in AD Agent.
11029	WARN	RADIUS	Unsupported RADIUS packet type	A RADIUS packet having an invalid packet type has been received. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly.
11030	WARN	RADIUS	Pre-parsing of the RADIUS packet failed	An invalid RADIUS packet has been received. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly.
11031	WARN	RADIUS	RADIUS packet type is not a valid Request	A RADIUS response packet has been received when a RADIUS request packet was expected. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly.
11036	ERROR	RADIUS	The Message-Authenticator RADIUS attribute is invalid.	A RADIUS packet having an invalid Message-Authenticator attribute has been received. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly. Make sure that the same RADIUS shared secret has been properly configured, both in the client device and in AD Agent.
11037	ERROR	RADIUS	Dropped accounting request received via unsupported port.	A RADIUS Accounting-Request packet was received via an unsupported UDP port number. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly.

Table B-1 AD Agent Log Messages (continued)

Message Code	Logging Level	Message Class	Message Text	Description
11038	ERROR	RADIUS	RADIUS Accounting-Request header contains invalid Authenticator field.	A RADIUS Accounting-Request packet having an invalid Authenticator field in its packet header has been received. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly. Make sure that the same RADIUS shared secret has been properly configured, both in the client device and in AD Agent.
11039	INFO	RADIUS	RADIUS authentication request rejected due to critical logging error	An internal logging-related error has been detected, possibly due to lack of sufficient available disk space.
11040	INFO	RADIUS	RADIUS accounting request dropped due to critical logging error.	An internal logging-related error has been detected, possibly due to lack of sufficient available disk space.
11050	WARN	RADIUS	RADIUS request dropped due to system overload	An internal logging-related error has been detected, possibly due to lack of sufficient available disk space.
11052	ERROR	RADIUS	Authentication request dropped due to unsupported port number	A RADIUS request packet was received via an unsupported UDP port number. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly.
11053	WARN	RADIUS	Invalid attributes in outgoing radius packet - possibly some attributes exceeded their size limit	An internal has caused an outgoing RADIUS response packet to be invalid, possibly because the size of the packet as a whole, or of one of its associated attributes, had exceeded the maximum limit.
11103	ERROR	RADIUS-Client	RADIUS-Client encountered error during processing flow	An internal error has been detected during the processing of an incoming RADIUS packet. Make sure that the client device is compatible with AD Agent, has been configured properly, and is functioning properly. Make sure that the same RADIUS shared secret has been properly configured, both in the client device and in AD Agent.

Table B-1 AD Agent Log Messages (continued)

Message Code	Logging Level	Message Class	Message Text	Description
11213	WARN	Dynamic-Authorization	No response received from Network Access Device: lost communication with notification subscriber	AD Agent has not received any acknowledgement packet (positive or negative) from a client device to which it has previously sent a CoA-Request packet. AD Agent assumes that communication with this client device has been lost, and will set its status to 'Out-Of-Sync'.
11214	WARN	Dynamic-Authorization	An invalid response received from Network Access Device: lost communication with notification subscriber	AD Agent has received an invalid response from a client device to which it has previously sent a CoA-Request packet. AD Agent assumes that communication with this client device has been lost, and will set its status to 'Out-Of-Sync'.
32006	WARN	Logging	Could not log to critical logger	An internal logging-related error has been detected, possibly due to lack of sufficient available disk space.
32016	FATAL	Logging	System reached low disk space limit	Sufficient disk space is not available.

APPENDIX C

Windows Application Event Logs

This appendix summarizes the events that the AD Agent software logs to the Windows Application Log (in addition to the Customer Log messages), under the following situations:

- The AD Agent itself (actually, its internal "watchdog" functionality) has been started or stopped because of installation or uninstallation of the AD Agent software, or because of a reboot of the AD Agent machine.
- The internal “AD Observer” and “RADIUS Server” components of the AD Agent have been manually started or stopped, using the **adactrl** command.
- The internal “AD Observer” and “RADIUS Server” components of the AD Agent have been automatically stopped or restarted, after the watchdog functionality of the AD Agent software has detected a crash or a serious error in one or more of these processes.

You can view these events using the Windows “Event Viewer” tool, which is available in the following location:

Event Viewer (Local) > Applications and Services Log > Cisco AD Agent

All these events have the following attributes and values:

- Source—Cisco AD Agent
- Level—Information
- Task Category—None

Table C-1 lists the event ID, message text, and description for these events.

Table C-1 Windows Application Event Logs

Event ID	Message Text	Description
10	Watchdog Service Was Started	The internal watchdog service of the AD Agent has been started. You will usually see this message immediately after installation of the AD Agent, or after a reboot of the AD Agent machine.
11	Watchdog Service Was Shutdown	The internal watchdog service of the AD Agent has been stopped. You will usually see this message as part of uninstallation of the AD Agent. It can also happen if the Cisco AD Agent in the Windows Services panel was manually stopped or restarted.
20	C:\\IBF\\watchdog\\radiusServer.bat Was Started	The “RADIUS Server” subcomponent of the AD Agent has either been manually started (using the **adactrl** command) or automatically restarted (after a crash or failure).
20	C:\\IBF\\watchdog\\adObserver.bat Was Started	The “AD Observer” subcomponent of the AD Agent has either been manually started (using the **adactrl** command) or automatically restarted (after a crash or failure).
21	C:\\IBF\\watchdog\\radiusServer.bat Was Shutdown	The “RADIUS Server” subcomponent of the AD Agent has crashed, or has been stopped after a failure was detected.
21	C:\\IBF\\watchdog\\adObserver.bat Was Shutdown	The “AD Observer” subcomponent of the AD Agent has crashed, or has been stopped after a failure was detected.
21	rt_daemon.exe Was Shutdown	The “RADIUS Server” subcomponent of the AD Agent has been manually stopped (using the **adactrl** command).
21	ADObserver.exe Was Shutdown	The “AD Observer” subcomponent of the AD Agent has been manually stopped (using the **adactrl** command).
22	C:\\IBF\\watchdog\\radiusServer.bat Failed To Start	The “RADIUS Server” subcomponent of the AD Agent has failed to start properly.
22	C:\\IBF\\watchdog\\adObserver.bat Failed To Start	The “AD Observer” subcomponent of the AD Agent has failed to start properly.

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

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

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

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.