Hewlett Packard
Enterprise

HPE Security ArcSight Connectors

SmartConnector for IBM AIX Realtime Audit File (Legacy)

Configuration Guide

November 30, 2016

Configuration Guide

SmartConnector for IBM AIX Realtime Audit File (Legacy)

November 30, 2016

Warranty

The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise Development LP shall not be liable for technical or editorial omissions contained herein. The information contained herein is subject to change without notice. The network information used in the examples in this document (including IP addresses and hostnames) is for illustration purposes only. Hewlett Packard Enterprise Development LP products are highly flexible and function as you configure them. The accessibility, integrity, and confidentiality of your data is your responsibility. Implement a comprehensive security strategy and follow good security practices. This document is confidential.

Restricted Rights Legend

Follow this link to see a complete statement of Hewlett Packard Enterprise Development LP copyrights, trademarks and acknowledgements:
https://www.protect724.hpe.com/docs/DOC-13026.

Revision History

Date	Description
11/30/2016	Updated installation procedure for setting preferred IP address mode.
08/30/2016	Marked this connector as Legacy. Use the SmartConnector for AIX Audit Syslog for continuing version support.
09/30/2013	Removed support for AIX 5.3.
05/15/2012	Added new installation procedure.
02/15/2012	Added support for AIX version 7.1.
09/30/2011	Updated descriptions of AIX Audit File and AIX Realtime Audit File connectors.
09/24/2010	Support for version 6.1 was added in the 5.0.1.5994 SmartConnector release.
06/25/2010	Added troubleshooting information regarding slow performance in HostName lookup.
02/11/2010	Added support for CEF File transport.

Contents

SmartConnector for IBM AIX Realtime Audit File (Legacy)

This guide provides information for installing the SmartConnector for IBM AIX Realtime Audit File and for configuring the device for log event collection. This SmartConnector is supported on AIX platforms only. AIX Audit versions 6.1 and 7.1 are supported. For the latest version support, use the SmartConnector for IBM AIX Audit Syslog.

Product Overview

The purpose of the AIX PR auditing system is to record instances of access by subjects to objects and to allow detection of any (repeated) attempts to bypass the protection mechanism and any misuses of privileges.

There are three SmartConnectors for AIX Audit:

- **AIX Audit Log Realtime (Legacy) (this SmartConnector)**
  This realtime connector processes the audit logs in real time. During connector installation, it is configured with a script that collects the events and translates them into human-readable form before forwarding them to the ArcSight ESM Manager. This script must be launched and executed on the AIX system itself; therefore, remote event collection is not possible with this connector.

- **IBM AIX Audit Log File (Legacy)**
  This SmartConnector does not process AIX audit logs in real time. During the connector installation, it is configured with a temporary folder that it monitors continuously for any audit log files deposited. Any files deposited are processed immediately and the events are sent to the ArcSight ESM Manager.

- **IBM AIX Audit Syslog**
  This connector supersedes the two AIX Audit file connectors and provides the latest AIX version support.

Set Up Auditing

Event Collection

Information collection encompasses logging the selected auditable events. The audit logger is responsible for constructing the complete audit record, consisting of the audit header, which contains information common to all events (such as the name of the event, the user responsible , the time and return status of the event) and the *audit trail*, which contains event-specific information. The audit logger appends each successive record to the kernel audit trail, which can be written in either (or both) BIN and STREAM modes.

Default Script

A default script `$ARCSIGHT_HOME/current/bin/aixauditpr/tailauditlogs` is provided with the SmartConnector. You also can specify your own script from another location in order to filter or select specific events you want to monitor. The `auditselect` command lets you filter events from the raw binary log; however, this filtered data still must be piped to the same command as in the ArcSight-provided script:

```
/usr/sbin/auditpr -v -h e,l,R,t,c,p,P
```

This same set of command line switches to the `auditpr` command must be used to produce the audit logs in the human-readable format the SmartConnector understands.

See the *AIX Security* manual for specific information about configuring AIX auditing for your AIX version.

Select Audit Events

Auditing lets you detect activities that might compromise the security of your system. When performed by an unauthorized user, the following activities violate system security and are candidates for an audit:

- Engaging in activities in the Trusted Computing Base
- Authenticating users
- Accessing the system
- Changing the configuration of the system
- Circumventing the auditing system
- Initializing the system
- Installing programs
- Modifying accounts
- Transferring information into or out of the system

The audit system does not have a default set of events to be audited. You must select events or event classes according to your needs.

To audit an activity, identify the command or process that initiates the audit event and ensure that the event is listed in the `/etc/security/audit/events` file for your system. Then add the event either to an appropriate class in the `/etc/security/audit/config` file, or to an object stanza in the `/etc/security/audit/objects` file.

See the `/etc/security/audit/events` file on your system for the list of audit events and trail formatting instructions. For a description of how audit event formats are written and used, see the **auditpr** command.

Group into Audit Classes

After you have selected the events to audit, combine similar events into audit classes. These audit classes are defined in the classes stanza of the `/etc/security/audit/config` file. Then assign audit classes to users. Some typical audit classes are:

General

Events that alter the state of the system and change user authentication. Audit attempts to circumvent system access controls.

Objects

Write access to security configuration files.

Kernel

Events in the kernel class are generated by the process management functions of the kernel.

An example of a stanza in the `/etc/security/audit/config` file follows.

```
classes:
     general = USER_SU,PASSWORD_Change,FILE_Unlink,FILE_Link,FILE-Rename
     system = USER_Change,GROUP_Change,USER_Create,GROUP_Create
     init = USER_Login,USER_Logout
```

Assigning Audit Events to an Object

Assign the audit events to an object (data or executable file) by adding a stanza for that file to the `/etc/secutity/audit/objects` file. To get all audit events, specify the ALL class; however, be aware that with this option, a huge amount of data will be generated.

Select an Audit Data Collection Method

The audit data collection method you choose depends upon how you intend to use the audit data. If you need long-term storage of a large amount of data, select BIN collection. If you want to process the data as it is collected, select STREAM collection. If you need both long-term storage and immediate processing, select both methods.

In the `/etc/security/audit/config` file, configure whether you want to use BIN collection, STREAM collection, or both methods. Use a separate file system for audit data to ensure that audit data does not compete with other data for file space.

To configure BIN collection:

1. Enable to BIN mode collection by setting `binmode = on` in the Start stanza.

2. Edit the Binmode stanza to configure the bins and trail, and specify the path of the file containing the BIN mode back-end processing commands. The default file for back-end commands is the `/etc/security/audit/bincmds` file.

3. Make sure the audit bins are large enough for your needs and set the `freespace` parameter accordingly to receive an alert if the file system is filing up.

4. Include the shell commands that process the audit bins in an audit pipe in the `/etc/security/audit/bincmds` file.

To configure STREAM collection:

1. Enable the STREAM mode collection by setting `streammode = on` in the Start stanza.

2. Edit the Streammode stanza to specify the path to the file containing the streammode processing commands. The default file containing this information is the `/etc/security/audit/streamcmds` file.

3 Include the shell commands that process the stream records in an audit pipe in the `/etc/security/audit/streamcmds` file.

Enable the Audit Subsystem

When you have finished making any necessary changes to the configuration files, you can use the `audit start` command to enable the audit subsystem. You can use the `audit shutdown` command to deactivate the audit subsystem.

When the audit start or audit shutdown command is executed, the configuration information is reset and the audit logs are flushed to the streams. When this happens, the SmartConnector must be restarted.

The auditpr Command

The auditpr command reads audit records, in bin or stream format, from standard input and sends formatted records to standard output.

The output format is determined by flags that are selected. If you specify the `-m` flag, a message is displayed before each heading. Use the `-h` flag to change the default fields and the `-v` flag to append an audit trail. The auditpr command searches the local `/etc/passwd` file to convert user and group IDs to names.

Values that can be used with the -h flag to select fields are as follows:

Value	Description
e	The audit event.
l	The user's login name.
R	The audit status.
t	The time the record was written.
c	The command name.
r	The real user name.
p	The process ID.
P	The ID of the parent process.
T	The kernel thread ID (local to the process; different process can contain threads with the same thread ID).
h	The name of the host that generated the audit record. If there is no CPU ID in the audit record, the value **none** is used. If there is no matching entry for the CPU ID in the audit record, the 16-character value for the CPU ID is used instead.

The `e`, `l`, `R`, `t`, and `c` flags are used by default.

Sample Event File

An example of the `/etc/security/audit/event` file:

```
[#]/etc/security/audit]> cat events
....
auditpr:
```

...other rows precede

```
*kernel proc events

*      fork()
       PROC_Create = printf "forked child process %d"

*      exit()
       PROC_Delete = printf "exited child process %d"

*      exec()
       PROC_Execute = printf "euid: %d egid: %d epriv: %x:%x name %s"
```

... other rows follow

For examples of audit trails, see the **/etc/security/audit/events** file where the audit trail formats are defined.

Example of auditpr Command

```
[#][/] /usr/sbin/audit pr -v < audit/trail
event            login     status            time            command
--------------------------------------------------------------------
FS_Chdir         root      OK       Tue Oct 05 12:58:26 2004 ksh
FILE_Unlink      root      OK       Tue Oct 05 12:59:03 2004 vi
FILE_Unlink      root      OK       Tue Oct 05 12:59:12 2004 vi
FS_Chdir         root      OK       Tue Oct 05 12:59:34 2004 ksh
FS_Chdir         root      OK       Tue Oct 05 12:59:37 2004 ksh
FILE_Unlink      root      OK       Tue Oct 05 12:59:40 2004 vi
FILE_Unlink      root      OK       Tue Oct 05 12:59:59 2004 vi
CRON_Start       root      OK       Tue Oct 05 13:00:00 2004 cron
FS_Chdir         root      OK       Tue Oct 05 13:00:00 2004 cron
FILE_Unlink      root      OK       Tue Oct 05 13:00:02 2004 vi
FILE_Unlink      root      OK       Tue Oct 05 13:00:04 2004 vi
FILE_Unlink      root      OK       Tue Oct 05 13:02:38 2004 vi
FILE_Unlink      root      OK       Tue Oct 05 13:02:44 2004 vi
FILE_Unlink      root      OK       Tue Oct 05 13:02:44 2004 vi
TCPIP_connect    root      OK       Tue Oct 05 13:20:15 2004 telnetd
FILE_Write       root      OK       Tue Oct 05 13:20:15 2004 telnetd
```

Example of Config File for BIN Mode

```
[#][/etc/security/audit]> head -20 config

start:
        binmode = on
            streammode = on
```

```
bin:
            trail = /audit/trail
            bin1 = /audit/bin1
            bin2 = /audit/bin2
            binsize = 10240
            cmds = /etc/security/audit/bincmds
            freespace = 65536

...
[#][/etc/security/audit]> cat /etc/secrutiy/audit/bincmds
/usr/sbin/auditcat -p -o $trail $bin
[p630n02][/etc/security/audit]>
```

Example of Config File for STREAM Mode

```
[#][/etc/security/audit]> cat config

start:
        binmode = on
        streammode = on

stream:
        cmds = /etc/security/audit/streamcmds
...
[#][/etc/security/audit]> cat  /etc/security/audit/streamcmds
/usr/sbin/auditstream | auditpr -v > /audit/stream.out &
[#][/ETC/SECURITY/AUDIT]>
```

Example of a Generic Audit Log Scenario

This example was derived from the AIX Security Guide, Setting Up Auditing chapter.

In this example, assume that a SYSADMIN wants to use the audit subsystem to monitor a large multi-user server system. No direct integration into an IDS is performed, all audit records will be inspected manually for irregularities. Only a few essential audit events are recorded, to keep the amount of generated data to a manageable size.

The audit events that are considered for event detection are:

FILE_WRITE
We want to know about file writes to configuration files, so this event will be used with all files in the /etc tree.

PROC_SetUserIDs	All changes of user ids.
AUD_Bin_Def	Audit bin configuration.
USER_SU	The su command.
PASSWORD_Change	The passwd command.

CRON_JobAdd	New cron jobs.
AT_JobAdd	New at jobs.
USER_Login	All logins.
PORT_Locked	All locks on terminals because of too many invalid attempts.

The following is an example of how to generate a generic audit log:

1 Set up a list of critical files to be monitored for changes, such as all files in /etc, and configure them for FILE_Write events in the **objects** file as follows:

```
find /etc -type f | awk '{printf("%s:\n\tw = FILE_Write\n\n",$1)}' >>
/etc/security/audit/objects
```

2 Use the auditcat command to set up BIN mode auditing. The **/etc/security/audit/bincmds** file is similar to the following:

```
/usr/sbin/auditcat -p -o $trail $bin
```

3 Edit the **/etc/security/audit/config** file and add a class for the events in which we are interested. List all existing users and specify the custom class for them.

```
start:
       binmode = on
       streammode = off

bin:
       cmds = /etc/security/audit/bincmds
       trail = /audit/trail
       bin1 = /audit/bin1
       bin2 = /audit/bin2
       binsize = 100000
       freespace = 100000

classes:
       custom = FILE-Write,PROC_SetUser,AUD_Bin_Def,AUD_Lost_Rec,
                USER_SU,PASSWORD_Change,CRON_JobAdd,AT_JobAdd,USER_Login,
                PORT_Locked

users:
       root = custom
       afx = custom
       ...
```

4 Add the **custom** audit class to the **/usr/lib/security/mkuser.default** file, so that new IDs will automatically have the correct audit call associated:

```
user:
       auditclasses = custom
       pgrp = staff
       groups = staff
       shell = /usr/bin/ksh
       home = /home/$USER
```

5 Create a new file system named **/audit** by using SMIT or the **crfs** command. The file system should be large enough to hold the two bins and a large audit trail.

6 Run the **audit start** command option and examine the **/audit** file. You should see the two bin files and an empty **trail** file initially. After you have used the system for a while, you should have audit records in the **trail** file that can be read with:

```
auditpr -hhelpPRrTc -v | more
```

This example uses only a few events. To see all events, you could specify the classname **ALL** for all users. This action will generated large amounts of data. You might want to add all events related to user changes and privilege changes to your **custom** class.

Example of Real-Time File Modification Monitoring

The following example can be used to monitor file access to critical files in real time:

1 Set up a list of critical files to be monitored for changes; for example, all files in **/etc**, and configure them for **FILE_Write** events in the **objects** file.

```
find /etc -type f | awk '{printf("%s:\n\tw = FILE_Write\n\n",$1)}' >>
/etc/security/audit/objects
```

2 Set up stream auditing to list all file writes. (This example lists all file writes to the console, but in using the ArcSight SmartConnector in a production environment, you would want to have a backend that sends the events into an Intrusion Detection System.) The **/etc/security/audit/streamcmds** file is similar to the following:

```
/usr/sbin/auditstream | /usr/sbin/auditselect -e "event == FILE_Write" |
auditpr -hhelpPRtTc -v > /dev/console &
```

3 Set up STREAM mode auditing in **/etc/security/audit/config**; add a class for the file write events and configure all users that should be audited with that class:

```
start:
       binmode = off
       streammode = on

stream:
       cmds = /etc/security/audit/streamcmds

classes:
       filemon = FILE_Write
```

```
users:
        root = filemon
        afx = filemon
        ...
```

4 Now run **audit start**. All FILE_Write events are displayed on the console.

When the audit start or audit shutdown command is executed, the configuration information is reset and the audit logs are flushed to the streams. When this happens, the SmartConnector must be restarted.

AIX Configuration Files

AIX configuration files you may need to access include:

File	Description
/usr/sbin/auditselect	Specifies the path of the auditselect command.
/etc/rc	Contains the system initialization commands.
/etc/security/audit/config	Contains audit system configuration information.
/etc/security/audit/events	Contains the audit events of the system.
/etc/security/audit/objects	Contains audit events for audit objects (files).
/etc/security/audit/bincmds	Contains auditbin backend commands.
/etc/security/audit/streamcmds	Contains auditstream commands.

See the *AIX Security* manual for specific information about configuring AIX auditing for your AIX version.

Install the SmartConnector

The following sections provide instructions for installing and configuring your selected SmartConnector.

Connector Appliance/ArcSight Management Center supports mounting for Network File System (NFS) and CIFS (Windows) shares. When you install this connector on one of these devices, establish a CIFS mount on the device before adding the connector. Provide this share name during connector configuration. For more information, see **Remote File Systems** in the Connector Appliance or ArcSight Management Center Administrator's Guide.

Prepare to Install Connector

Before you install any SmartConnectors, make sure that the ArcSight products with which the connectors will communicate have already been installed correctly (such as ArcSight ESM or ArcSight Logger). This configuration guide takes you through the installation process with **ArcSight Manager (encrypted)** as the destination.

For complete product information, read the *Administrator's Guide* as well as the *Installation and Configuration* guide for your ArcSight product before installing a new SmartConnector. If you are adding a connector to the ArcSight Management Center, see the *ArcSight Management Center Administrator's Guide* for instructions, and start the installation procedure at "Set Global Parameters (optional)" or "Select Connector and Add Parameter Information."

Before installing the SmartConnector, be sure the following are available:

- Local access to the machine where the SmartConnector is to be installed
- Administrator passwords

Install Core Software

Unless specified otherwise at the beginning of this guide, this SmartConnector can be installed on all ArcSight supported platforms; for the complete list, see the *SmartConnector Product and Platform Support* document, available from the HPE SSO and Protect 724 sites.

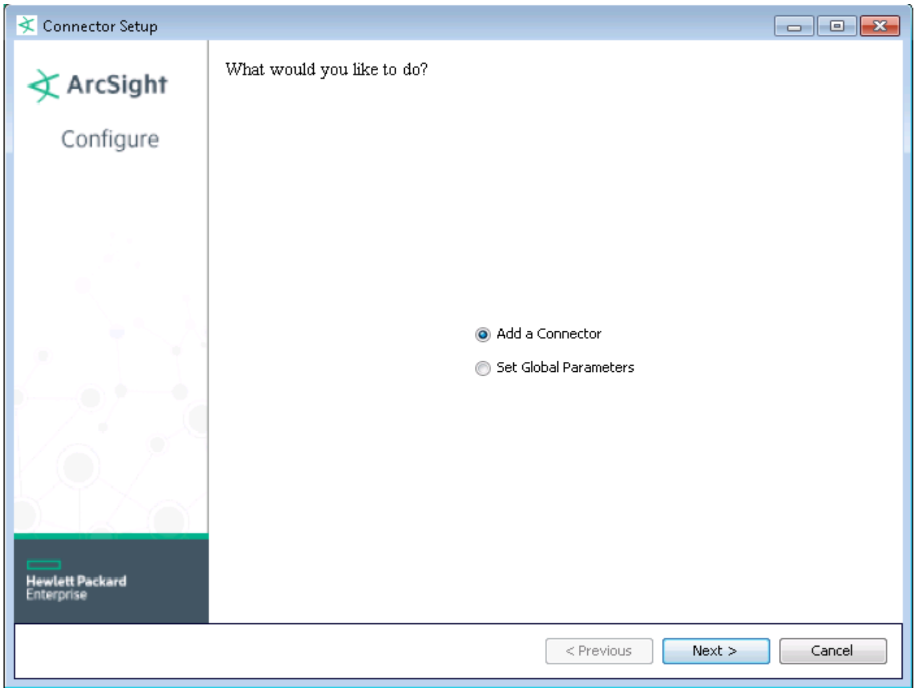
1. Download the SmartConnector executable for your operating system from the HPE SSO site.

2. Start the SmartConnector installation and configuration wizard by running the executable.

   Follow the wizard through the following folder selection tasks and installation of the core connector software:

   Introduction
   Choose Install Folder
   Choose Shortcut Folder
   Pre-Installation Summary
   Installing...

3. When the installation of SmartConnector core component software is finished, the following window is displayed:



What would you like to do?

● Add a Connector
○ Set Global Parameters

Set Global Parameters (optional)

If you choose to perform any of the operations shown in the following table, do so before adding your connector. After installing core software, you can set the following parameters:

Global Parameter	Setting
Set FIPS mode	Set to 'Enable' to enable FIPS compliant mode. To enable FIPS Suite B Mode, see the SmartConnector User Guide under "Modifying Connector Parameters" for instructions. Initially, this value is set to 'Disable'.
Set Remote Management	Set to 'Enable' to enable remote management from ArcSight Management Center. When queried by the remote management device, the values you specify here for enabling remote management and the port number will be used. Initially, this value is set to 'Disable'.
Remote management listener port	The remote management device will listen to the port specified in this field. The default port number is 9001.
Preferred IP Version	If both 'IPv4' and'IPv6' IP addresses are available for the local host (the machine on which the connector is installed), you can choose which version is preferred. Otherwise, you will see only one selection. When both values are present, the initial setting is 'IPv4'.

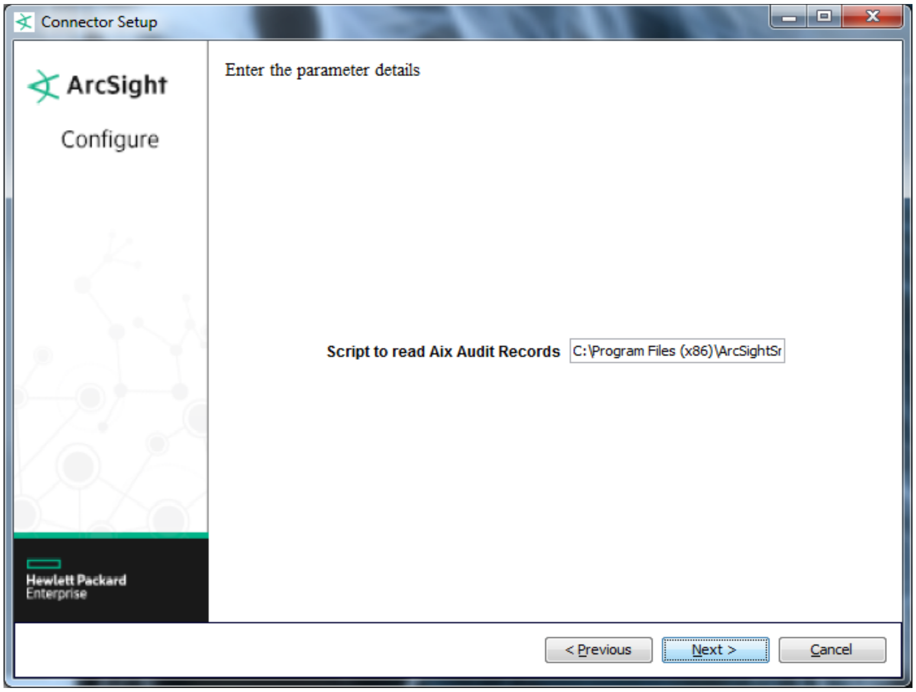
After making your selections, click **Next**. A summary screen is displayed. Review the summary of your selections and click **Next**. Click **Continue** to return to the "Add a Connector" window. Continue the installation procedure with "Select Connector and Add Parameter Information."

Select Connector and Add Parameter Information

1. Select **Add a Connector** and click **Next**. If applicable, you can enable FIPS mode and enable remote management later in the wizard after SmartConnector configuration.

2. Select **IBM AIX Realtime Audit File (Legacy)** and click **Next**.

3. Enter the required SmartConnector parameters to configure the SmartConnector, then click **Next**.



Enter the parameter details

Script to read Aix Audit Records: C:\Program Files (x86)\ArcSightSr

Parameter	Description
Script to read Aix Audit Records	Script to be executed to read AIX audit records in human readable form. See "Default Script" in "Setting Up Auditing" for more information.

Select a Destination

1. The next window asks for the destination type; make sure **ArcSight Manager (encrypted)** is selected and click **Next**. (For information about this destination or any of the other destinations listed, see the *ArcSight SmartConnector User Guide*.)

2. Enter values for the **Manager Host Name**, **Manager Port**, **User** and **Password** required parameters. This is the same ArcSight user name and password you created during the ArcSight Manager installation. Click **Next**.

3. Enter a name for the SmartConnector and provide other information identifying the connector's use in your environment. Click **Next**. The connector starts the registration process.

4. The certificate import window for the ArcSight Manager is displayed. Select **Import the certificate to the connector from destination** and click **Next**. (If you select **Do not import the certificate to connector from destination**, the connector installation will end.) The certificate is imported and the **Add connector Summary** window is displayed.

Complete Installation and Configuration

1. Review the **Add Connector Summary** and click **Next**. If the summary is incorrect, click **Previous** to make changes.

2 The wizard now prompts you to choose whether you want to run the SmartConnector as a stand-alone process or as a service. If you choose to run the connector as a stand-alone process, select **Leave as a standalone application**, click **Next**, and continue with step 5.

3 If you chose to run the connector as a service, with **Install as a service** selected, click **Next**. The wizard prompts you to define service parameters. Enter values for **Service Internal Name** and **Service Display Name** and select **Yes** or **No** for **Start the service automatically**. The **Install Service Summary** window is displayed when you click **Next**.

4 Click **Next** on the summary window.

5 To complete the installation, choose **Exit** and Click **Next**.

For some SmartConnectors, a system restart is required before the configuration settings you made take effect. If a **System Restart** window is displayed, read the information and initiate the system restart operation.

Save any work on your computer or desktop and shut down any other running applications (including the ArcSight Console, if it is running), then shut down the system.

For instructions about upgrading the connector or modifying parameters, see the *SmartConnector User Guide*.

Run the SmartConnector

SmartConnectors can be installed and run in stand-alone mode, on Windows platforms as a Windows service, or on UNIX platforms as a UNIX daemon, depending upon the platform supported. On Windows platforms, SmartConnectors also can be run using shortcuts and optional Start menu entries.

If the connector is installed in stand-alone mode, it must be started manually and is not automatically active when a host is restarted. If installed as a service or daemon, the connector runs automatically when the host is restarted. For information about connectors running as services or daemons, see the *ArcSight SmartConnector User Guide*.

To run all SmartConnectors installed in stand-alone mode on a particular host, open a command window, go to `$ARCSIGHT_HOME\current\bin` and run: `arcsight connectors`

To view the SmartConnector log, read the file `$ARCSIGHT_HOME\current\logs\agent.log`; to stop all SmartConnectors, enter `Ctrl+C` in the command window.

Device Event Mapping to ArcSight Fields

The following section lists the mappings of ArcSight data fields to the device's specific event definitions. See the *ArcSight Console User's Guide* for more information about the ArcSight data fields.

IBM AIX Audit Field Mappings

ArcSight ESM Field	Device-Specific Field
Agent (Connector) Severity	Medium when Device Severity = FAIL, FAIL_ACCESS, FAIL_PRIV, FAIL_DAC, FAIL_AUTH; Low when Device Severity = OK
Application Protocol	protocol
Destination Host Name	host

ArcSight ESM Field	Device-Specific Field
Destination Port	port
Destination Process Name	pid
Destination Service Name	cmd or Program
Destination User Name	user
Device Custom Number 1	File Descriptor
Device Custom Number 2	Parent PID
Device Custom Number 3	Physical Volume Index
Device Custom String 1	ACL
Device Custom String 2	Group
Device Custom String 3	Owner
Device Custom String 4	Reason or Error Code
Device Custom String 5	PCL
Device Custom String 6	Volume Group ID
Device Event Class Id	event
Device Product	'AIX Audit PR'
Device Receipt Time	Time
Device Severity	Status
Device Vendor	'IBM'
Event Outcome	status
File ID	ID
File Path	One of (File Name, Path)
File Permission	Mode
File Size	File size
Message	message
Name	event
Priority	Priority
Source Process Name	process
Source Service Name	command
Source User ID	uid
Source User Name	login
Type	type

Troubleshooting

Why does the server hang when I initiate a reboot?

With AIX 5.3 and later, if you follow the configuration information, the server will hang when you initiate a reboot. This problem is not with the SmartConnector, but with auditd on AIX. Add the following to the startup script:

```
/usr/sbin/audit start 1>&- 2>&-
```

The system will reboot without any issue and the audit mechanism will work.

Why is performance slow or the server hangs when performing HostName lookup?

For complete information about this problem, see IBM Support & downloads, Slow performance or hang in HostName lookup due to high DNS traffic. Information in this troubleshooting section has been derived from that material.

Cause: The problem could be lookup issues between IPv6 and IPv4 in releases of the Java 2 SDK 1.4 if the JVM performs both IPv6 and IPv4 queries. If the Domain Name System (DNS) server is not configured to handle IPv6 queries, the application (such as './runagentsetup.sh' for the connector) may issue an unknown host exception. If the DNS is not configured to handle IPv6 queries properly, the application must wait for the IPv6 query to time out.

Solution:

- If you are using IPv4 and IPv6 addresses in your environment, it could mean that the DNS server is not set up properly. In that case, contact your network administrator to look into the problem.
- If you are certain that you will communicate only with IPv4 addresses, you can set the following Java property in the AIX machine where the connector is installed:

  ```
  -Djava.net.preferIPv4Stack=true
  ```

If you add this property, the implication is that the application will not be able to communicate with IPv6 hosts.