

Chapter 21

Integrating IP Address Managers with the SAE

This chapter describes how you can integrate IP address managers, such as a Dynamic Host Configuration Protocol (DHCP) server or a RADIUS server, into an SDX-managed network so that the SAE is notified about subscriber events. It contains the following sections:

- Overview of IP Address Manager Integration on page 403
- Installing Monitoring Agent on page 405
- Configuring Monitoring Agent on page 405
- Managing Monitoring Agent on page 408

Overview of IP Address Manager Integration

You use the Monitoring Agent application with the event notification method of logging in subscribers and creating subscriber sessions. You can use event notification when you integrate devices into the SDX network that do not notify the SAE about subscriber events, such as when a subscriber logs in or when the address assignment is terminated.

For example, you can use monitoring agent in a cable network. When events occur between the IP address manager and the cable modem termination system (CMTS) device or PacketCable Multimedia Specification (PCMM) device driver, Monitoring Agent creates event notifications on the IP address manager that are delivered to the SAE using the event notification application programming interface (API).

For information about event notification in the PCMM network, see *SDX Solutions Guide, Chapter 4, Providing Premium Services in a PCMM Environment*.

For information about event notification with other third-party network devices, see *SDX Integration Guide, Chapter 1, Integrating Third-Party Network Devices in the SDX Network*.

The Monitoring Agent application monitors DHCP or RADIUS messages for DHCP or RADIUS servers running on the same host as Monitoring Agent and generates subscriber events. Monitoring Agent intercepts messages on every available interface unless configured to do otherwise in the property file.

The Monitoring Agent application must run on every server host that can allocate IP addresses to subscribers. Monitoring Agent is stateless and cannot synchronize the current set of subscribers when there is a failure. If events are missed because of a software or network failure, the overall state recovers when DHCP leases are renewed or RADIUS interim updates are sent. For example, missed ipUp events become effective when the affected lease is renewed or the next interim update is sent, and missed ipDown events time out when the lease expires or after the configured RADIUS time to live.

Monitoring DHCP Messages

When Monitoring Agent is intercepting DHCP messages, it captures every UDP packet that is received or sent on UDP port 67 (BOOTP/DHCP server).

Monitoring Agent processes messages for the following DHCP message types:

- DHCPACK—Sent from the server to the client when a lease is acknowledged. The Monitoring Agent application translates the client IP address and IP address lease time into an ipUp event.
- DHCPNAK—Sent from the server to the client when a lease is not renewed or the client configuration is wrong. The Monitoring Agent application translates the client IP address into an ipDown event.
- DHCPRELEASE—Sent from the client to the server when the client cancels the lease. The Monitoring Agent application translates the client IP address into an ipDown event.

All other DHCP messages are ignored.

Monitoring RADIUS Messages

When Monitoring Agent is intercepting RADIUS accounting messages, it captures every UDP packet that is sent to the RADIUS accounting port (1813 is the default port).

Monitoring Agent processes messages for the following RADIUS attributes:

- Acct-Status-Type (RADIUS attribute [40])—Start and interim update events are translated into ipUp events. Stop events are translated into ipDown events.
- Framed-Ip-Address (RADIUS attribute [8])—The IP address identifies the notified interface.
- Acct-Session-Id (RADIUS attribute [44])—The accounting session ID is set as the EA_SESSION_ID attribute of the event notification.
- NAS-Port-Id (RADIUS attribute [87])—If present, the NAS port ID is set as the EA_NAS_PORT_ID attribute of the event notification.

The RADIUS client must send interim update accounting requests with a known frequency because the Monitoring Agent application cannot keep the state of logged subscriber sessions. To allow for lost messages, you might set the timeout value for ipUp notifications to a value that is larger than the interim update interval. For example, setting the timeout value to twice the interim update interval allows for one lost message.

Installing Monitoring Agent

You must manually install the UMCmagt package on the server host to deploy the Monitoring Agent application.

```
pkgadd -d /cdrom/cdrom0/solaris UMCmagt
```

For information about installing Monitoring Agent, see *Chapter 1, Installing the SDX Applications*.

Configuring Monitoring Agent

This section lists the configuration tasks for Monitoring Agent.

- Configuring Properties on page 405
- Configuring NIC Proxy on page 407

Configuring Properties

The properties for Monitoring Agent determine the behavior of the application. The default values allow the Monitoring Agent application to operate, but you can specify different timeout values, device names, or RADIUS ports.

To configure properties for Monitoring Agent:

1. On the server host, log in as `root` or as an authorized nonroot admin user.
2. Verify that Monitoring Agent is not running. (See *Displaying Monitoring Agent Status* on page 409 and *Stopping Monitoring Agent* on page 408.)
3. With a text editor, edit the `/opt/UMC/monAgent/etc/ma_default.properties` file. Change the properties using the following descriptions to configure the Monitoring Agent application.
4. Save the file.
5. Start Monitoring Agent for the changes to take effect. (See *Starting Monitoring Agent* on page 408.)

MonAgent.capture.devices

- Space-delimited list of devices where packets are captured. When this list is empty, packets on all available interfaces are captured.
- Value—Text string in the format < interfaceName > < interfaceName >
 - < interfaceName > identifies the network interface on the host where the Monitoring Agent application is running.
- Default—Empty
- Example—dmfe0 dmfe1

MonAgent.capture.pool

- Maximum number of concurrent event handlers.
- Value—Integer in the range 0-2147483647
- Default—8

MonAgent.timeout

- Time to keep an event handler alive for reuse.
- Value—Number of seconds in the range 0-2147483647
- Default—300

MonAgent.event.timeout

- Time to wait before discarding failed events.
- Value—Number of seconds in the range 0-2147483647
- Default—300

MonAgent.event.retry_time

- Time to wait before retrying failed events.
- Value—Number of seconds in the range 0-2147483647
- Default—30

MonAgent.dhcp.packet.forward

- Controls the attachment of the whole packet to the notification.
- Value
 - true—Enables the attachment of the packet.
 - false—Disables the attachment of the packet.
- Default—true

MonAgent.dhcp.enable

- Controls the monitoring of DHCP messages.
- Value
 - true—Enables the monitoring of DHCP messages.
 - false—Disables the monitoring of DHCP messages.
- Default—true

MonAgent.radius.enable

- Controls the monitoring of RADIUS messages.
- Value
 - true—Enables the monitoring of RADIUS messages.
 - false—Disables the monitoring of RADIUS messages.
- Default—true

MonAgent.radius.port

- UDP port on which RADIUS accounting messages are expected.
- Value—Integer; valid port number in the range 1–65535
- Default—1813

MonAgent.ttl

- Time to wait for a detected IP address.
- Value—Number of seconds in the range 0-2147483647
- Guidelines—This timeout value should be larger than interim update interval; we recommend twice the value.
- Default—1800

Configuring NIC Proxy

To configure a NIC proxy for the Monitoring Agent application, see *SDX Network Guide: SAE, Juniper Networks Routers, and NIC, Chapter 7, Configuring Applications to Communicate with an SAE*.

Managing Monitoring Agent

The Monitoring Agent application must be running on the same host as each DHCP server or RADIUS server that can allocate IP addresses to subscribers.

To manage Monitoring Agent, you can perform these tasks:

- Starting Monitoring Agent on page 408
- Stopping Monitoring Agent on page 408
- Displaying Monitoring Agent Status on page 409
- Cleaning Monitoring Agent Logs on page 409

Starting Monitoring Agent

Before you start Monitoring Agent, you must do the following:

1. Install Monitoring Agent as described in *Installing Monitoring Agent* on page 405.
2. Configure Monitoring Agent as described in *Configuring Monitoring Agent* on page 405.

To start Monitoring Agent:

1. On the Monitoring Agent host, log in as `root`.
2. Start Monitoring Agent from its installation directory.

```
/opt/UMC/monAgent/etc/monAgent start
```

The system responds with a start message. If Monitoring Agent is already running, the system responds with a warning message.

Stopping Monitoring Agent

Before you reconfigure Monitoring Agent, you must manually stop it.

To stop Monitoring Agent:

1. On the Monitoring Agent host, log in as `root`.
2. Stop Monitoring Agent from its installation directory.

```
/opt/UMC/monAgent/etc/monAgent stop
```

The system responds with a stop message. If Monitoring Agent is not running when you issue the command, the system responds with a warning message.

Displaying Monitoring Agent Status

To display the Monitoring Agent status:

1. On the Monitoring Agent host, log in as `root`.
2. Display the status from the Monitoring Agent installation directory.

```
/opt/UMC/monAgent/etc/monAgent status
```

The system responds with a status message.

Cleaning Monitoring Agent Logs

To delete the log files for Monitoring Agent:

1. On the Monitoring Agent host, log in as `root`.
2. Delete the log files from the Monitoring Agent installation directory.

```
/opt/UMC/monAgent/etc/monAgent clean
```

By using the `stdout` and `stderr` options, you can clean the log files for the Monitoring Agent application and delete the persistent data that the agent writes to files or devices. See *SDX Monitoring and Troubleshooting Guide, Chapter 2, Configuring Logging for SDX Components* for more information.

