Chapter 5

Configuring the SAE for a PCMM Environment with the SRC CLI

This chapter describes how to configure the SAE for a PacketCable Multimedia Specification (PCMM) environment with the SRC CLI.

- To use the C-Web interface to configure the SAE for a PCCM environment, see *SRC-PE C-Web Interface Configuration Guide, Chapter 11, Configuring the SAE for a PCMM Environment with the C-Web Interface*.

- For information about setting up SAE groups, see *SRC-PE Getting Started Guide, Chapter 21, Setting Up an SAE with the SRC CLI*.

- For information about setting up a community manager, see *Setting Up SAE Communities with SRC CLI* on page 55.

Topics in this chapter include:

- Configuring the SAE for a Cable Network Environment with SRC CLI on page 52

- Configuring the SAE to Manage PCMM Devices with SRC CLI on page 53

- Setting Up SAE Communities with SRC CLI on page 55

- Configuring SAE Properties for the Event Notification API with SRC CLI on page 57

- Configuring Record-Keeping Server Peers for Plug-Ins with SRC CLI on page 58

- Configuring PCMM Record-Keeping Server Plug-Ins with SRC CLI on page 59

- Configuring CMTS-Specific RKS Plug-Ins with SRC CLI on page 61

Configuring the SAE for a Cable Network Environment with SRC CLI

The tasks to configure the SAE for a cable network environment are:

1. Configure the SAE to manage PCMM devices.

    Configuring the SAE to Manage PCMM Devices with SRC CLI on page 53.

2. Configure the session store.

    See *SRC-PE Network Guide, Chapter 2, Configuring the SAE with the SRC CLI.*

3. Set up SAE communities.

    See Setting Up SAE Communities with SRC CLI on page 55.

4. (Optional) Configure SAE properties for the event notification API.

    See Configuring SAE Properties for the Event Notification API with SRC CLI on page 57 (if you are using an external address manager).

5. (Optional) Configure record-keeping server peers for plug-ins.

    See Configuring Record-Keeping Server Peers for Plug-Ins with SRC CLI on page 58 (if you are using the RKS plug-in).

6. (Optional) Configure PCMM record-keeping server plug-ins.

    See Configuring PCMM Record-Keeping Server Plug-Ins with SRC CLI on page 59 (if you are using the SAE's embedded policy server).

7. (Optional) Configure CMTS-specific RKS plug-ins.

    See Configuring CMTS-Specific RKS Plug-Ins with SRC CLI on page 61.

In addition to configuring the SAE, you need to:

1. Configure the CMTS device in the directory (if you are using the SAE's embedded policy server).

    See Adding Objects for CMTS Devices with the SRC CLI on page 63.

2. Configure the NIC (if you are using assigned IP subscribers).

    See *Chapter 8, Using the NIC Resolver in a PCMM Environment.*

3. Enable the Common Open Policy Service (COPS) interface on the CMTS device. See the documentation for your CMTS device for information about how to do this.

Configuring the SAE to Manage PCMM Devices with SRC CLI

The SAE connects to the PCMM device by using a COPS over TCP connection. The PCMM device driver controls this connection.

Use the following configuration statements to configure the SAE to manage CMTS devices:

```
shared sae configuration driver pcmm {
    keepalive-interval keepalive-interval;
    tcp-connection-timeout tcp-connection-timeout;
    application-manager-id application-manager-id;
    message-timeout message-timeout;
    cops-message-maximum-length cops-message-maximum-length;
    cops-message-read-buffer-size cops-message-read-buffer-size;
    cops-message-write-buffer-size cops-message-write-buffer-size;
    sae-community-manager sae-community-manager;
    disable-full-sync disable-full-sync;
    disable-pcmm-i03-policy disable-pcmm-i03-policy;
    session-recovery-retry-interval session-recovery-retry-interval;
    element-id element-id;
    default-rks-plug-in default-rks-plug-in;
}
```

To configure the SAE to manage CMTS devices:

1. From configuration mode, access the configuration statement that configures the PCMM driver. In this sample procedure, the PCMM device driver is configured in the west-region group.

    user@host# **edit shared sae group west-region configuration driver pcmm**

2. Configure the interval between keepalive messages sent from the COPS client (the PCMM device) to the COPS server (the SAE).

    [edit shared sae group west-region configuration driver pcmm]
    user@host# **set keepalive-interval** keepalive-interval

3. Configure the timeout for opening a TCP connection to the PCMM device.

    [edit shared sae group west-region configuration driver pcmm]
    user@host# **set tcp-connection-timeout** tcp-connection-timeout

4. When this SAE is configured as the application manager, configure the identifier of the application manager.

    [edit shared sae group west-region configuration driver pcmm]
    user@host# **set application-manager-id** application-manager-id

5. Configure the time that the COPS server (the SAE) waits for a response to COPS requests from the COPS client (the PCMM device). Change this value only if a high number of COPS timeout events appear in the error log.

    [edit shared sae group west-region configuration driver pcmm]
    user@host# **set message-timeout** message-timeout

6. Configure the maximum length of a COPS message.

   [edit shared sae group west-region configuration driver pcmm]
   user@host# **set cops-message-maximum-length** *cops-message-maximum-length*

7. Configure the buffer size for receiving COPS messages from the COPS client.

   [edit shared sae group west-region configuration driver pcmm]
   user@host# **set cops-message-read-buffer-size** *cops-message-read-buffer-size*

8. Configure the buffer size for sending COPS messages to the COPS client.

   [edit shared sae group west-region configuration driver pcmm]
   user@host# **set cops-message-write-buffer-size** *cops-message-write-buffer-size*

9. Configure the name of the community manager that manages PCMM driver communities. Active SAEs are selected from this community.

   [edit shared sae group west-region configuration driver pcmm]
   user@host# **set sae-community-manager** *sae-community-manager*

10. Enable or disable state synchronization with PCMM policy servers.

    [edit shared sae group west-region configuration driver pcmm]
    user@host# **set disable-full-sync** *disable-full-sync*

11. Enable or disable the SAE to send classifiers to the router that comply with PCMM IO3. Disable this option if your network deployment has CMTS devices that do not support PCMM IO3.

    [edit shared sae group west-region configuration driver pcmm]
    user@host# **set disable-pcmm-i03-policy** *disable-pcmm-i03-policy*

12. Configure the time between attempts by the SAE to restore service sessions that are being recovered in the background when state synchronization completes with a state-data-incomplete error.

    [edit shared sae group west-region configuration driver pcmm]
    user@host# **set session-recovery-retry-interval** *session-recovery-retry-interval*

13. (Optional) Configure the unique identifier that the SAE uses to identify itself when it originates in record-keeping server (RKS) events.

    [edit shared sae group west-region configuration driver pcmm]
    user@host# **set element-id** *element-id*

14. (Optional) Specify the name of the default RKS plug-in to which the SAE sends events for CMTS devices.

    [edit shared sae group west-region configuration driver pcmm]
    user@host# **set default-rks-plug-in** *default-rks-plug-in*

15. (Optional) Verify your PCMM driver configuration.

    ```
    [edit shared sae group west-region configuration driver pcmm]
    user@host# show
    keepalive-interval 45;
    ```

```
tcp-connection-timeout 5;
application-manager-id 1;
message-timeout 120000;
cops-message-maximum-length 204800;
cops-message-read-buffer-size 3000;
cops-message-write-buffer-size 3000;
sae-community-manager PcmmCommunityManager;
disable-full-sync true;
disable-pcmm-i03-policy true;
session-recovery-retry-interval 3600000;
element-id 1;
default-rks-plug-in rksTracking;
```

Setting Up SAE Communities with SRC CLI

You can configure the following for SAE communities:

- Define the members of an SAE community by adding the IP addresses of SAEs in the community to the virtual router object of the network device in the directory.

  See Creating Virtual Routers for the CMTS Device with the SRC CLI on page 64.

- Configure parameters for the SAE community manager.

  See Configuring the SAE Community Manager on page 55.

- Specify the name of the community manager with the **set sae-community-manager** option in the PCMM driver configuration.

  See Configuring the SAE to Manage PCMM Devices with SRC CLI on page 53.

- If there is a firewall in the network, configure the firewall to allow SAE messages through.

Configuring the SAE Community Manager

Use the following configuration statements to configure the SAE community manager that manages PCMM device communities:

shared sae configuration external-interface-features *name* CommunityManager {
    keepalive-interval *keepalive-interval*;
    threads *threads*;
    acquire-timeout *acquire-timeout*;
    blackout-time *blackout-time*;
}

To configure the community manager:

1. From configuration mode, access the configuration statements for the community manager. In this sample procedure, west_region is the name of the SAE group, and sae_mgr is the name of the community manager.

   user@host# **edit shared sae group west-region configuration external-interface-features sae_mgr CommunityManager**

2. Specify the interval between keepalive messages sent from the active SAE to the passive members of the community.

   [edit shared sae group west-region configuration external-interface-features sae_mgr CommunityManager]
   user@host# **set keepalive-interval** *keepalive-interval*

3. Specify the number of threads that are allocated to manage the community. You generally do not need to change this value.

   [edit shared sae group west-region configuration external-interface-features sae_mgr CommunityManager]
   user@host# **set threads** *threads*

4. Specify the amount of time an SAE waits for a remote member of the community when it is acquiring a distributed lock. You generally do not need to change this value.

   [edit shared sae group west-region configuration external-interface-features sae_mgr CommunityManager]
   user@host# **set acquire-timeout** *acquire-timeout*

5. Specify the amount of time that an active SAE must wait after it shuts down before it can try to become the active SAE of the community again.

   [edit shared sae group west-region configuration external-interface-features sae_mgr CommunityManager]
   user@host# **set blackout-time** *blackout-time*

6. (Optional) Verify the configuration of the SAE community manager.

   ```
   [edit shared sae group west-region configuration external-interface-features
   sae_mgr CommunityManager]
   user@host# show
   CommunityManager {
     keepalive-interval 30;
     threads 5;
     acquire-timeout 15;
     blackout-time 30;
   }
   ```

Related Topics

■ *SRC-PE Getting Started Guide, Chapter 21, Setting Up an SAE with the SRC CLI.*

Configuring SAE Properties for the Event Notification API with SRC CLI

Use the following configuration statements to configure properties for the Event Notification API:

```
shared sae configuration external-interface-features name EventAPI {
    retry-time retry-time;
    retry-limit retry-limit;
    threads threads;
}
```

To configure properties for the Event Notification API:

1. From configuration mode, access the configuration statements for the Event Notification API. In this sample procedure, west-region is the name of the SAE group, and event_api is the name of the Event API configuration.

   user@host# **edit shared sae group west-region configuration external-interface-features event_api EventAPI**

2. Specify the amount of time between attempts to send events that could not be delivered.

   [edit shared sae group west-region configuration external-interface-features event_api EventAPI]
   user@host# **set retry-time** *retry-time*

3. Specify the number of times an event fails to be delivered before the event is discarded.

   [edit shared sae group west-region configuration external-interface-features event_api EventAPII]
   user@host# **set retry-limit** *retry-limit*

4. Specify the number of threads allocated to process events.

   [edit shared sae group west-region configuration external-interface-features event_api EventAPI]
   user@host# **set threads** *threads*

5. (Optional) Verify the configuration of the Event Notification API properties.

   ```
   [edit shared sae group west-region configuration external-interface-features
   event_api EventAPI]
   user@host# show
   EventAPI {
     retry-time 300;
     retry-limit 5;
     threads 5;
   }
   ```

Related Topics

- *SRC-PE Getting Started Guide, Chapter 21, Setting Up an SAE with the SRC CLI.*

Configuring Record-Keeping Server Peers for Plug-Ins with SRC CLI

An RKS peer is an instance of an RKS. A PCMM environment has a primary RKS and optionally a secondary RKS. The primary RKS is mandatory, and you assign the RKS as primary by configuring it as the default peer in the RKS plug-in. The secondary RKS is optional, and it is an RKS peer that is not configured as the default peer. If you define multiple nondefault peers, one of them is randomly chosen to be the secondary RKS.

RKS peers are configured in the peer group for each PCMM RKS plug-in instance. To create an RKS peer group:

Use the following configuration statements to configure an RKS peer group.

shared sae configuration plug-ins name *name* pcmm-rks peer-group *name* {
    server-address *server-address*;
    server-port *server-port*;
}
To configure an RKS peer group:

1. From configuration mode, access the configuration statements for RKS plug-ins. In this sample procedure, west-region is the name of the SAE group, and rksPlugin is the name of the plug-in and rksPeer is the name of the peer group.

    user@host# **edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks peer-group rksPeer**

2. Specify the IP address of the RKS server to which the SAE sends accounting data.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks peer-group rksPeer]
    user@host# **set server-address** *server-address*

3. Specify the port used for sending accounting packets.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks peer-group rksPeer]
    user@host# **set server-port** *server-port*

4. (Optional) Verify your configuration.

    ```
    [edit shared sae group west-region configuration plug-ins name rksPlugin
    pcmm-rks peer-group rksPeer]
    user@host# show
    server-address 10.10.3.60;
    server-port 1812;
    ```

Related Topics

- *SRC-PE Getting Started Guide, Chapter 21, Setting Up an SAE with the SRC CLI.*

Configuring PCMM Record-Keeping Server Plug-Ins with SRC CLI

Use the following configuration statements to configure an RKS plug-in.

```
shared sae configuration plug-ins name name pcmm-rks {
    load-balancing-mode (failover | roundRobin);
    failback-timer failback-timer;
    retry-interval retry-interval;
    maximum-queue-length maximum-queue-length;
    bind-address bind-address;
    udp-port udp-port;
    feid-mso-data feid-mso-data;
    feid-mso-domain-name feid-mso-domain-name;
    trusted-element;
    default-peer default-peer;
}
```

To configure an RKS plug-in:

1. From configuration mode, access the configuration statements for RKS plug-ins. In this sample procedure, west-region is the name of the SAE group, and rksPlugin is the name of the plug-in.

    user@host# **edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks**

2. Specify the mode for load-balancing RKSs.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set load-balancing-mode** (failover | roundRobin)

3. Specify if and when the SAE attempts to fail back to the default peer.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set failback-timer** *failback-timer*

4. Specify the time the SAE waits for a response from an RKS before it resends the packet.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set retry-interval** *retry-interval*

5. Specify the maximum number of unacknowledged messages that the plug-in receives from the RKS before it discards new messages.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set maximum-queue-length** *maximum-queue-length*

6. (Optional) Specify the source IP address that the plug-in uses to communicate with the RKS.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set bind-address** *bind-address*

7. (Optional) Specify the source UDP port or a pool of ports that the plug-in uses to communicate with the RKS.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set udp-port** *udp-port*

8. (Optional) Specify the multiple service operator (MSO)—defined data in the financial entity ID (FEID) attribute, which is included in event messages.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set feid-mso-data** *feid-mso-data*

9. (Optional) Specify the MSO domain name in the FEID attribute that uniquely identifies the MSO for billing and settlement purposes.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set feid-mso-domain-name** *feid-mso-domain-name*

10. (Optional) When the SAE is running as a policy server—which means that the SAE sends event messages directly to the RKS—enable the SAE as a trusted network element.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set trusted-element**

11. Specify the name of the primary RKS peer to which the SAE sends accounting packets.

    See Configuring Record-Keeping Server Peers for Plug-Ins with SRC CLI on page 58.

    [edit shared sae group west-region configuration plug-ins name rksPlugin pcmm-rks]
    user@host# **set default-peer** *default-peer*

12. (Optional) Verify your RKS plug-in configuration.

```
[edit shared sae group west-region configuration plug-ins name rksPlugin
pcmm-rks]
user@host> show
load-balancing-mode failover;
failback-timer -1;
retry-interval 3000;
maximum-queue-length 10000;
feid-mso-domain-name abcd.com;
trusted-element;
default-peer radius01;
```

13. (Optional) Specify an RKS plug-in for specific CMTS devices.

See Configuring CMTS-Specific RKS Plug-Ins with SRC CLI on page 61.

Related Topics

- *SRC-PE Getting Started Guide, Chapter 21, Setting Up an SAE with the SRC CLI.*

Configuring CMTS-Specific RKS Plug-Ins with SRC CLI

You can configure an RKS plug-in for specific CMTS devices. When there are events for the CMTS device, the SAE sends the events to the specified plug-in.

Use the following configuration statement to assign a CMTS-specific RKS plug-in.

shared sae configuration driver pcmm cmts-specific-rks-plug-ins *name* {
    rks-plug-in *rks-plug-in*;
}

To configure a CMTS-specific RKS plug-in:

1. From configuration mode, access the configuration statements for RKS plug-ins. In this sample procedure, west-region is the name of the SAE group, and cmtsPlugin is the name of the plug-in assignment.

   user@host# **edit shared sae group west-region configuration driver pcmm cmts-specific-rks-plug-ins cmtsPlugin**

2. Specify the name of the CMTS-specific RKS plug-in.

   [edit shared sae group west-region configuration driver pcmm
   cmts-specific-rks-plug-ins cmtsPlugin]
   user@host# **set rks-plug-in** *rks-plug-in*

3. (Optional) Verify your configuration.

```
[edit shared sae group west-region configuration driver pcmm
cmts-specific-rks-plug-ins cmtsPlugin]
user@host# show
rks-plug-in rksPlugin;
```

Related Topics

- *SRC-PE Getting Started Guide, Chapter 21, Setting Up an SAE with the SRC CLI.*