



---

# Third-Party Network Devices with Scripting Service in the SRC Network



Published: 2014-12-10

Juniper Networks, Inc.  
1194 North Mathilda Avenue  
Sunnyvale, California 94089  
USA  
408-745-2000  
www.juniper.net

Copyright © 2014, Juniper Networks, Inc. All rights reserved.

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

*Third-Party Network Devices with Scripting Service in the SRC Network*  
Copyright © 2014, Juniper Networks, Inc.  
All rights reserved.

The information in this document is current as of the date on the title page.

#### YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

#### END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

# Table of Contents

	About the Documentation . . . . .	ix
	Documentation and Release Notes . . . . .	ix
	Supported Platforms . . . . .	ix
	Documentation Conventions . . . . .	ix
	Documentation Conventions . . . . .	x
	Documentation Feedback . . . . .	xii
	Requesting Technical Support . . . . .	xii
	Self-Help Online Tools and Resources . . . . .	xiii
	Opening a Case with JTAC . . . . .	xiii
<b>Part 1</b>	<b>Overview</b>	
<b>Chapter 1</b>	<b>Overview of Scripting Based Third-Party Network Device Integration . . . . .</b>	<b>3</b>
	Integrating Network Devices into the SRC Network Overview . . . . .	3
	SAE Communities . . . . .	3
	Storing Session Data . . . . .	4
	Using Script Services to Provision Third-Party Devices . . . . .	4
	Logging In Subscribers and Creating Sessions . . . . .	5
	Assigned IP Subscribers . . . . .	5
	Login Interactions with Assigned IP Subscribers . . . . .	5
	Event Notification from an IP Address Manager . . . . .	6
	Login with Event Notification . . . . .	7
<b>Part 2</b>	<b>Configuration</b>	
<b>Chapter 2</b>	<b>Configuration Overview . . . . .</b>	<b>11</b>
	Configuration Tasks for Integrating Third-Party Network Devices (SRC CLI) . . . . .	11
	Configuration Tasks for Integrating Third-Party Network Devices (C-Web Interface) . . . . .	12
<b>Chapter 3</b>	<b>Configuration Tasks for Integrating Scripting Based Third-Party Network Devices . . . . .</b>	<b>13</b>
	Setting Up Script Services . . . . .	13
	Adding Objects for Network Devices (SRC CLI) . . . . .	14
	Adding Virtual Router Objects (SRC CLI) . . . . .	15
	Adding Objects for Network Devices (C-Web Interface) . . . . .	16
	Adding a Router Object . . . . .	16
	Adding Virtual Router Objects . . . . .	17
	Setting Up SAE Communities (SRC CLI) . . . . .	17
	Configuring the SAE Community Manager . . . . .	17
	Specifying the Community Manager in the SAE Device Driver . . . . .	18

	Setting Up SAE Communities (C-Web Interface) . . . . .	19
	Configuring the SAE Community Manager . . . . .	19
	Specifying the Community Manager in the SAE Device Driver . . . . .	20
	Configuring SAE Properties for the Event Notification API (SRC CLI) . . . . .	20
	Configuring SAE Properties for the Event Notification API (C-Web Interface) . . . . .	21
	Developing Router Initialization Scripts for Network Devices and Juniper Networks	
	Routers . . . . .	21
	Interface Object Fields . . . . .	22
	Required Methods . . . . .	23
	Example: Router Initialization Script . . . . .	24
	Copying Initialization Scripts to the C Series Controller . . . . .	24
	Specifying Initialization Scripts on the SAE (SRC CLI) . . . . .	25
	Specifying Initialization Scripts on the SAE (C-Web Interface) . . . . .	25
	Using SNMP to Retrieve Information from Network Devices . . . . .	26
	Configuring Global SNMP Communities in the SRC Software (SRC CLI) . . . . .	26
	Configuring Global SNMP Communities in the SRC Software (C-Web Interface) . . . . .	27
	Using the NIC Resolver in Environments That Have Third-Party Devices (SRC CLI) . . . . .	27
	Using the NIC Resolver in Environments That Have Third-Party Devices (C-Web Interface) . . . . .	28
<b>Part 3</b>	<b>Administration</b>	
<b>Chapter 4</b>	<b>Monitoring Third-Party Network Devices with Scripting Service . . . . .</b>	<b>31</b>
	Viewing Information About SAE Device Drivers (SRC CLI) . . . . .	31
	Viewing Information About Device Drivers (C-Web Interface) . . . . .	32
	Viewing SNMP Information for Routers and Devices (SRC CLI) . . . . .	33
	Viewing Statistics for Device Drivers (SRC CLI) . . . . .	34
	Viewing SNMP Statistics for Devices (C-Web Interface) . . . . .	35
	Viewing Statistics for Specific Device Drivers (SRC CLI) . . . . .	36
	Viewing SNMP Statistics for Specific Devices (C-Web Interface) . . . . .	37
<b>Part 4</b>	<b>Troubleshooting</b>	
<b>Chapter 5</b>	<b>Troubleshooting Procedures . . . . .</b>	<b>41</b>
	Collecting Data with the Activity Monitor (SRC CLI) . . . . .	41
	Collecting Data with the Activity Monitor (C-Web Interface) . . . . .	42
<b>Part 5</b>	<b>Index</b>	
	Index . . . . .	45

# List of Figures

Part 1	Overview
Chapter 1	<b>Overview of Scripting Based Third-Party Network Device Integration . . . . . 3</b>
	Figure 1: SAE Community . . . . . 3
	Figure 2: Login Interactions with Assigned IP Subscribers . . . . . 6
	Figure 3: Login Interactions with Event Notification Application . . . . . 7



# List of Tables

	<b>About the Documentation</b> .....	<b>ix</b>
	Table 1: Notice Icons .....	x
	Table 2: Notice Icons .....	xi
	Table 3: Text Conventions .....	xi
<b>Part 2</b>	<b>Configuration</b>	
<b>Chapter 3</b>	<b>Configuration Tasks for Integrating Scripting Based Third-Party Network Devices</b> .....	<b>13</b>
	Table 4: Router Initialization Scripts .....	22
	Table 5: Exported Fields .....	23





# About the Documentation

- [Documentation and Release Notes on page ix](#)
- [Supported Platforms on page ix](#)
- [Documentation Conventions on page ix](#)
- [Documentation Feedback on page xii](#)
- [Requesting Technical Support on page xii](#)

## Documentation and Release Notes

---

To obtain the most current version of all Juniper Networks<sup>®</sup> technical documentation, see the product documentation page on the Juniper Networks website at <http://www.juniper.net/techpubs/>.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <http://www.juniper.net/books>.

## Supported Platforms

---

For the features described in this document, the following platforms are supported:







- [C Series](#)

## Documentation Conventions

---

[Table 1 on page x](#) defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

## Documentation Conventions

Table 1 on page x defines the notice icons used in this guide. Table 3 on page xi defines text conventions used throughout this documentation.

Table 2: Notice Icons







Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 3: Text Conventions

Convention	Description	Examples
<b>Bold text like this</b>	<ul style="list-style-type: none"> <li>Represents keywords, scripts, and tools in text.</li> <li>Represents a GUI element that the user selects, clicks, checks, or clears.</li> </ul>	<ul style="list-style-type: none"> <li>Specify the keyword <b>exp-msg</b>.</li> <li>Run the <b>install.sh</b> script.</li> <li>Use the <b>pkgadd</b> tool.</li> <li>To cancel the configuration, click <b>Cancel</b>.</li> </ul>
<b>Bold text like this</b>	Represents text that the user must type.	<b>user@host# set cache-entry-age</b> <i>cache-entry-age</i>
Fixed-width text like this	Represents information as displayed on your terminal's screen, such as CLI commands in output displays.	<pre>nic-locators {   login {     resolution {       resolver-name /realms/         login/A1;       key-type LoginName;       value-type SaeId;     }   } }</pre>
Regular sans serif typeface	<ul style="list-style-type: none"> <li>Represents configuration statements.</li> <li>Indicates SRC CLI commands and options in text.</li> <li>Represents examples in procedures.</li> <li>Represents URLs.</li> </ul>	<ul style="list-style-type: none"> <li><b>system ldap server{</b> <b>stand-alone;</b></li> <li>Use the <b>request sae modify device failover</b> <b>command</b> with the <b>force</b> option</li> <li><b>user@host# ...</b></li> <li><a href="http://www.juniper.net/techpubs/software/management/sdx/api-index.html">http://www.juniper.net/techpubs/software/management/sdx/api-index.html</a></li> </ul>

Table 3: Text Conventions (*continued*)

<i>Italic sans serif typeface</i>	Represents variables in SRC CLI commands.	<code>user@host# set local-address local-address</code>
Angle brackets	In text descriptions, indicate optional keywords or variables.	Another runtime variable is <gfwif>.
Key name	Indicates the name of a key on the keyboard.	Press Enter.
Key names linked with a plus sign (+)	Indicates that you must press two or more keys simultaneously.	Press Ctrl + b.
<i>Italic typeface</i>	<ul style="list-style-type: none"> <li>Emphasizes words.</li> <li>Identifies book names.</li> <li>Identifies distinguished names.</li> <li>Identifies files, directories, and paths in text but not in command examples.</li> </ul>	<ul style="list-style-type: none"> <li>There are two levels of access: <i>user</i> and <i>privileged</i>.</li> <li><i>SRC-PE Getting Started Guide</i>.</li> <li><i>o=Users, o=UMC</i></li> <li>The <i>/etc/default.properties</i> file.</li> </ul>
Backslash	At the end of a line, indicates that the text wraps to the next line.	<code>Plugin.radiusAcct-1.class=\ net.juniper.smgmt.sae.plugin\ RadiusTrackingPluginEvent</code>
Words separated by the   symbol	Represent a choice to select one keyword or variable to the left or right of this symbol. (The keyword or variable may be either optional or required.)	<code>diagnostic   line</code>

## Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page at the Juniper Networks Technical Documentation site at <http://www.juniper.net/techpubs/index.html>, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at <https://www.juniper.net/cgi-bin/docbugreport/>.
- E-mail—Send your comments to [techpubs-comments@juniper.net](mailto:techpubs-comments@juniper.net). Include the document or topic name, URL or page number, and software version (if applicable).

## Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <http://www.juniper.net/support/warranty/>.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

## Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>
- Download the latest versions of software and review release notes: <http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <http://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum: <http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://tools.juniper.net/SerialNumberEntitlementSearch/>

## Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/>.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <http://www.juniper.net/support/requesting-support.html>.



## PART 1

# Overview

- [Overview of Scripting Based Third-Party Network Device Integration on page 3](#)





## CHAPTER 1

# Overview of Scripting Based Third-Party Network Device Integration

- [Integrating Network Devices into the SRC Network Overview on page 3](#)
- [Logging In Subscribers and Creating Sessions on page 5](#)

## Integrating Network Devices into the SRC Network Overview

---

You can integrate third-party routers and other network devices into your SRC network. The SAE provides a driver that you can use to integrate the SAE with a third-party device. This device driver uses the session store to store and replicate subscriber and service session data within a community of SAEs.

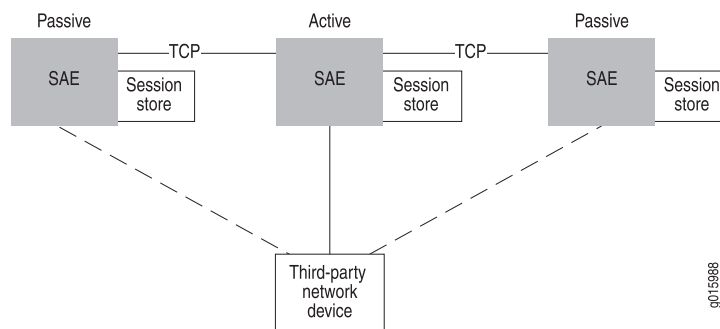
To log in subscribers to the SAE, you use assigned IP subscribers or event notification from an IP address manager.

To activate services and provision policies on the device, you use script services. You can also activate aggregate services for subscribers. However, you cannot activate normal services that require policies to be provisioned on the device.

## SAE Communities

For SAE redundancy in an SRC network, you can have a community of two or more SAEs. SAEs in a community are given the role of either active SAE or passive SAE. The active SAE manages the connection to the network device and keeps session data up to date within the community. [Figure 1 on page 3](#) shows a typical SAE community.

**Figure 1: SAE Community**



When an SAE starts, it negotiates with other SAEs to determine which SAE controls the network device. The SAE community manager and members of the community select the active SAE.

A passive SAE needs to take over as active SAE in any of the following cases:

- The active SAE shuts down. In this case, the active SAE notifies the passive SAEs, and one of the passive SAEs takes over as active SAE.
- A passive SAE does not receive a keepalive message from the active SAE within the keepalive interval. In this case, the passive SAE attempts to become the active SAE.

## Storing Session Data

To aid in recovering from an SAE failover, the SAE stores subscriber and service session data. When the SAE manages a network device, session data is stored in the SAE host's file system. The SRC component that controls the storage of session data on the SAE is called the session store. The session store queues data and then writes the data to session store files on the SAE host's disk. Once the data is written to disk, it can survive a server reboot.

For more information, see *Storing Subscriber and Service Session Data*.

## Using Script Services to Provision Third-Party Devices

You use script services to activate services and provision policies on third-party network devices. A script service is a service into which you can insert or reference a script. You write a script that will activate services and provision policies on the third-party device, and then you insert the script into the script service or reference the script in the service. When the SAE activates a service, it runs the script. The script provisions policies on the device using a means that the device supports. You can also include an interface in the script that causes the SAE to send authentication and tracking events when it activates, modifies, or deactivates a script service session.

The SAE core API includes two interfaces for creating a script:

- `ScriptService`—Defines a service provider interface (SPI) that the script service must implement. The implementation of the `ScriptService` interface activates, modifies, or deactivates the service.
- `ServiceSessionInfo`—Provides a callback interface into the SAE and provides information about the service session to the script service.

For information about the `ScriptService` interface and the `ServiceSessionInfo` interface, see the script service documentation in the SAE core API documentation on the Juniper Networks Web site at

<http://www.juniper.net/techpubs/software/management/src/api-index.html>

You can write the script in Java or Jython.

### Related Documentation

- [Logging In Subscribers and Creating Sessions on page 5](#)
- [Configuration Tasks for Integrating Third-Party Network Devices \(SRC CLI\) on page 11](#)

- [Adding Objects for Network Devices \(C-Web Interface\)](#) on page 16
- [Setting Up SAE Communities \(C-Web Interface\)](#) on page 19
- *Configuring the SAE Community Manager*

## Logging In Subscribers and Creating Sessions

---

You can use two mechanisms to obtain subscriber address requests and other information and to set up a pseudointerface on the network device. (You must choose one mechanism; you cannot mix them.)

1. Assigned IP subscriber. The SAE learns about a subscriber through subscriber-initiated activities, such as activating a service through the portal or through the SRC Web Services Gateway).

With this method, you use the assigned IP subscriber login type along with the network interface collector (NIC) to map IP addresses to the SAE.

2. Event notification from an IP address manager. The SAE learns about subscribers through notifications from an external IP address manager, such as a DHCP server or a RADIUS server.

With this method, you use the event notification application programming interface (API). The API provides an interface to the IP address manager, and lets the IP address manager notify the SAE of events such as IP address assignments.

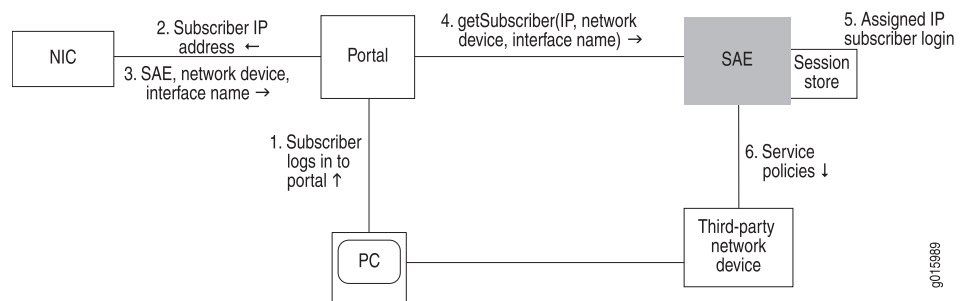
### Assigned IP Subscribers

With the assigned IP subscriber method of logging in subscribers and creating sessions, the SRC module uses IP address pools along with network information collector (NIC) resolvers to provide mapping of IP addresses to SAEs. You configure the static address pools or dynamically discovered address pools in the virtual router configuration for a network device. These pools are published in the NIC. The NIC maps subscriber IP addresses in requests received through the portal or SRC Web Services Gateway to the SAE that currently manages that network device.

### Login Interactions with Assigned IP Subscribers

This section describes login interactions for assigned IP subscribers. In the example shown in [Figure 2 on page 6](#), the subscriber activates a service through a portal. You could also have the subscriber activate a service through the SRC Web Services Gateway.

Figure 2: Login Interactions with Assigned IP Subscribers



The sequence of events for logging in and creating sessions for assigned IP subscribers is:

1. The subscriber logs in to the portal.
2. The portal sends the subscriber's IP address to the NIC.
3. Based on the IP address, the NIC looks up the subscriber's SAE, network device, and interface name, and returns this information to the portal.
4. The portal sends a get Subscriber message to the SAE. The message includes the subscriber's IP address, network device, and interface name.
5. The SAE creates an assigned IP subscriber and performs a subscriber login. Specifically, it:
  - a. Runs the subscriber classification script with the IP address of the subscriber. (Use the ASSIGNEDIP login type in subscriber classification scripts.)
  - b. Loads the subscriber profile.
  - c. Runs the subscriber authorization plug-ins.
  - d. Runs the subscriber tracking plug-ins.
  - e. Creates a subscriber session and stores the session data in the session store file.
6. The SAE pushes service policies for the subscriber session to the network device.

Because the SAE is not notified when the subscriber logs out, the assigned IP idle timer begins when no service is active. The SAE removes the interface subscriber session when the timeout period ends.

### Event Notification from an IP Address Manager

With the event notification method of logging in subscribers and creating subscriber sessions, the subscriber logs in to the network device and obtains an IP address through an address server, usually a DHCP server. The SAE receives notifications about the subscriber, such as the subscriber's IP address, from an event notification application that is installed on the DHCP server.

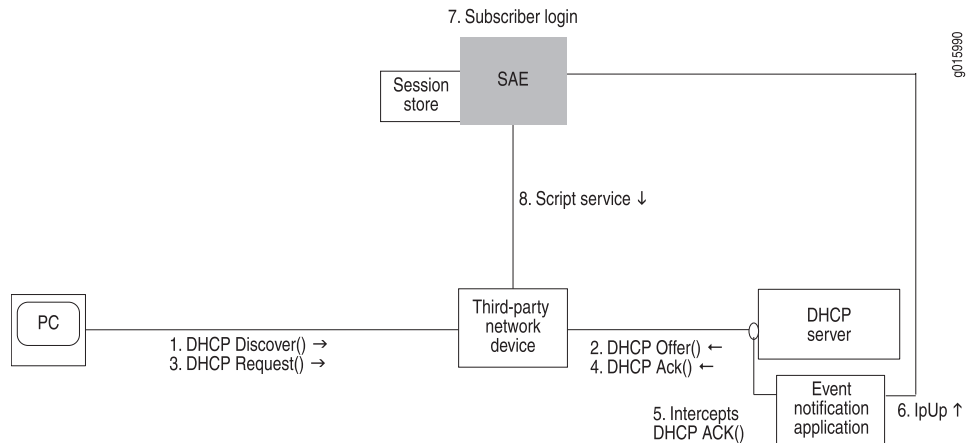
To use this method of logging in subscribers, you can use the event notification API to create the application that notifies the SAE when events occur between the DHCP server and the network device. You can also use Monitoring Agent, a sample application that

was created with the event notification API and that monitors DHCP or RADIUS messages for DHCP or RADIUS servers. See the *SRC PE Sample Applications Guide*.

## Login with Event Notification

This section describes login interactions by means of event notifications.

**Figure 3: Login Interactions with Event Notification Application**



The sequence of events for logging in subscribers and creating sessions is:

1. The DHCP client in the subscriber's computer sends a DHCP discover request to the DHCP server.
2. The DHCP server sends a DHCP offer to the subscriber's DHCP client.
3. The DHCP client sends a DHCP request to the DHCP server.
4. The DHCP server acknowledges the request by sending a DHCP Ack message to the DHCP client.
5. The event notification application that is running on the DHCP server intercepts the DHCP Ack message.
6. The event notification application sends an ipUp message to the SAE that notifies the SAE that an IP address is up.
7. The SAE performs a subscriber login. Specifically, it:
  - a. Runs the subscriber classification script.
  - b. Loads the subscriber profile.
  - c. Runs the subscriber authorization plug-ins.
  - d. Runs the subscriber tracking plug-ins.
  - e. Creates a subscriber session and stores the session in the session store file.
8. The SAE can start script services.

The ipUp event should be sent with a timeout set to the DHCP lease time. The DHCP server sends an ipUp event for each Ack message sent to the client. The SAE restarts the timeout each time it receives an ipUp event.

If the client explicitly releases the DHCP address (that is, it sends a DHCP release event), the DHCP server sends an ipDown event. If the client does not renew the address, the lease expires on the DHCP server and the timeout expires on the SAE.

**Related  
Documentation**

- [Integrating Network Devices into the SRC Network Overview on page 3](#)
- [Using the NIC Resolver in Environments That Have Third-Party Devices \(C-Web Interface\) on page 28](#)
- [Configuration Tasks for Integrating Third-Party Network Devices \(SRC CLI\) on page 11](#)
- [Configuring SAE Properties for the Event Notification API \(SRC CLI\) on page 20](#)
- [Adding Objects for Network Devices \(SRC CLI\) on page 14](#)
- [Setting Up Script Services on page 13](#)

## PART 2

# Configuration

- [Configuration Overview on page 11](#)
- [Configuration Tasks for Integrating Scripting Based Third-Party Network Devices on page 13](#)





## CHAPTER 2

# Configuration Overview

- Configuration Tasks for Integrating Third-Party Network Devices (SRC CLI) on page 11
- Configuration Tasks for Integrating Third-Party Network Devices (C-Web Interface) on page 12

## Configuration Tasks for Integrating Third-Party Network Devices (SRC CLI)

To integrate third-party devices into your SRC network, complete the following tasks:

- Write a script and add a script service that references the script.  
See “Setting Up Script Services” on page 13.
- Add objects for the devices.  
See “Adding Objects for Network Devices (SRC CLI)” on page 14.
- Configure an SAE community.  
See “Setting Up SAE Communities (SRC CLI)” on page 17.
- (Optional) Configure SAE properties for the Event Notification API if you are using the event notification method to log in subscribers.  
See “Configuring SAE Properties for the Event Notification API (SRC CLI)” on page 20.
- Configure the session store.  
See *Storing Subscriber and Service Session Data*.
- If you are using the event notification method to log in subscribers, integrate the SAE with an IP address manager. There are two ways to do so:
  - Use the event notification API to create an application that notifies the SAE when events occur between the DHCP server and the network device.  
See the event notification API documentation in the SAE CORBA remote API documentation on the Juniper Networks Web site at <http://www.juniper.net/techpubs/software/management/src/api-index.html>.
  - Use Monitoring Agent, a sample application that was created with the event notification API and that monitors DHCP or RADIUS messages for DHCP or RADIUS servers.  
See the *SRC PE Sample Applications Guide*.

- Related Documentation**
- [Configuration Tasks for Integrating Third-Party Network Devices \(C-Web Interface\) on page 12](#)
  - [Integrating Network Devices into the SRC Network Overview on page 3](#)
  - [Logging In Subscribers and Creating Sessions on page 5](#)

## Configuration Tasks for Integrating Third-Party Network Devices (C-Web Interface)

To integrate third-party devices into your SRC network, complete the following tasks:

- Write a script and add a script service that references the script.
- Add objects for the devices.  
See [“Adding Objects for Network Devices \(C-Web Interface\)” on page 16](#).
- Configure an SAE community.  
See [“Setting Up SAE Communities \(C-Web Interface\)” on page 19](#).
- (Optional) Configure SAE properties for the event notification API if you are using the event notification method to log in subscribers.  
See [“Configuring SAE Properties for the Event Notification API \(C-Web Interface\)” on page 21](#).
- Configure the session store.  
See *Storing Subscriber and Service Session Data*.
- If you are using the event notification method to log in subscribers, integrate the SAE with an IP address manager. There are two ways to do so:
  - Use the event notification API to create an application that notifies the SAE when events occur between the DHCP server and the network device.  
See the event notification API documentation in the SAE CORBA remote API documentation on the Juniper Networks Web site at <http://www.juniper.net/techpubs/software/management/src/api-index.html>
  - Use Monitoring Agent, an application that was created with the event notification API and that monitors DHCP or RADIUS messages for DHCP or RADIUS servers.  
See the *SRC Sample Applications Guide*.

- Related Documentation**
- [Configuration Tasks for Integrating Third-Party Network Devices \(SRC CLI\) on page 11](#)
  - [Using the NIC Resolver in Environments That Have Third-Party Devices \(C-Web Interface\) on page 28](#)
  - [Integrating Network Devices into the SRC Network Overview on page 3](#)

## CHAPTER 3

# Configuration Tasks for Integrating Scripting Based Third-Party Network Devices

- [Setting Up Script Services on page 13](#)
- [Adding Objects for Network Devices \(SRC CLI\) on page 14](#)
- [Adding Virtual Router Objects \(SRC CLI\) on page 15](#)
- [Adding Objects for Network Devices \(C-Web Interface\) on page 16](#)
- [Setting Up SAE Communities \(SRC CLI\) on page 17](#)
- [Setting Up SAE Communities \(C-Web Interface\) on page 19](#)
- [Configuring SAE Properties for the Event Notification API \(SRC CLI\) on page 20](#)
- [Configuring SAE Properties for the Event Notification API \(C-Web Interface\) on page 21](#)
- [Developing Router Initialization Scripts for Network Devices and Juniper Networks Routers on page 21](#)
- [Copying Initialization Scripts to the C Series Controller on page 24](#)
- [Specifying Initialization Scripts on the SAE \(SRC CLI\) on page 25](#)
- [Specifying Initialization Scripts on the SAE \(C-Web Interface\) on page 25](#)
- [Using SNMP to Retrieve Information from Network Devices on page 26](#)
- [Configuring Global SNMP Communities in the SRC Software \(SRC CLI\) on page 26](#)
- [Configuring Global SNMP Communities in the SRC Software \(C-Web Interface\) on page 27](#)
- [Using the NIC Resolver in Environments That Have Third-Party Devices \(SRC CLI\) on page 27](#)
- [Using the NIC Resolver in Environments That Have Third-Party Devices \(C-Web Interface\) on page 28](#)

## Setting Up Script Services

---

To set up script services:

1. Write a script that implements the ScriptService interface, a service provider interface (SPI) for the SAE.

See *Customizing Service Implementations*.

See the script service documentation in the SAE core API documentation on the Juniper Networks Web site at

<http://www.juniper.net/techpubs/software/management/src/api-index.html>

2. Add a script service that references the script.

See *SRC Script Services Overview*.

#### Related Documentation

- [Configuration Tasks for Integrating Third-Party Network Devices \(SRC CLI\) on page 11](#)
- [Copying Initialization Scripts to the C Series Controller on page 24](#)
- [Integrating Network Devices into the SRC Network Overview on page 3](#)
- [Logging In Subscribers and Creating Sessions on page 5](#)
- [Setting Up SAE Communities \(C-Web Interface\) on page 19](#)

## Adding Objects for Network Devices (SRC CLI)

---

For each network device that the SAE manages, add a router object and virtual router object.

Use the following configuration statements to add a router object:

```
shared network device name {  
  description description;  
  management-address management-address;  
  device-type (junose| junos| pcmm| third-party);  
  qos-profile [qos-profile...];  
}
```

To add a router object:

1. From configuration mode, access the statements that configure network devices. You must specify the name of a device with lowercase characters. This sample procedure uses `proxy_device` as the name of the router.

```
user@host# edit shared network device proxy_device
```

2. (Optional) Add a description for the router object.

```
[edit shared network device proxy_device]  
user@host# set description description
```

3. (Optional) Add the IP address of the router object.

```
[edit shared network device proxy_device]  
user@host# set management-address management-address
```

4. Set the type of device that you are adding to third-party.

```
[edit shared network device proxy_device]
```

```
user@host# set device-type third-party
```

- (Optional) Verify your configuration.

```
[edit shared network device proxy_device]
user@host# show
description "Third-party router";
management-address 192.168.9.25;
device-type third-party;
interface-classifier {
  rule rule-0 {
    script #;
  }
}
```

#### Related Documentation

- [Adding Objects for Network Devices \(C-Web Interface\) on page 16](#)
- [Configuration Tasks for Integrating Third-Party Network Devices \(SRC CLI\) on page 11](#)
- [Adding Virtual Router Objects \(SRC CLI\) on page 15](#)
- [Integrating Network Devices into the SRC Network Overview on page 3](#)
- [Logging In Subscribers and Creating Sessions on page 5](#)

## Adding Virtual Router Objects (SRC CLI)

Use the following configuration statements to add a virtual router:

```
shared network device name virtual-router name {
  sae-connection [sae-connection ...];
  snmp-read-community snmp-read-community;
  snmp-write-community snmp-write-community;
  scope [scope...];
  tracking-plug-in [tracking-plug-in...];
}
```

To add a virtual router:

- From configuration mode, access the statements for virtual routers. You must specify the name of a device with lowercase characters. This sample procedure uses `proxy_device` as the name of the router object. For third-party devices, use the name default for the virtual router.

```
user@host# edit shared network device proxy_device virtual-router default
```

- Specify the addresses of SAEs that can manage this router. This step is required for the SAE to work with the router.

```
[edit shared network device proxy_device virtual-router default]
user@host# set sae-connection [sae-connection ...]
```

To specify the active SAE and the redundant SAE, enter an exclamation point (!) after the hostname or IP address of the connected SAE. For example:

```
[edit shared network device proxy_device virtual-router default]
```

```
user@host# set sae-connection [sae1! sae2!]
```

- (Optional) Specify an SNMP community name for SNMP read-only operations for this virtual router.

```
[edit shared network device proxy_device virtual-router default]
user@host# set snmp-read-community snmp-read-community
```

- (Optional) Specify an SNMP community name for SNMP write operations for this virtual router.

```
[edit shared network device proxy_device virtual-router default]
user@host# set snmp-write-community snmp-write-community
```

- (Optional) Specify service scopes assigned to this virtual router. The scopes are available for subscribers connected to this virtual router for selecting customized versions of services.

```
[edit shared network device proxy_device virtual-router default]
user@host# set scope [ scope ...]
```

- (Optional) Specify the plug-ins that track interfaces that the SAE manages on this virtual router.

```
[edit shared network device proxy_device virtual-router default]
user@host# set tracking-plugin [ tracking-plugin ...]
```

- (Optional) Verify your configuration.

```
[edit shared network device proxy_device virtual-router default]
user@host# show
sae-connection 10.8.221.45;
snmp-read-community *****;
snmp-write-community *****;
scope POP-Toronto;
tracking-plugin flexRadius;
```

#### Related Documentation

- [Adding Objects for Network Devices \(C-Web Interface\) on page 16](#)
- [Adding Objects for Network Devices \(SRC CLI\) on page 14](#)
- [Integrating Network Devices into the SRC Network Overview on page 3](#)

## Adding Objects for Network Devices (C-Web Interface)

For each network device that the SAE manages, add a router object and virtual router object.

- [Adding a Router Object on page 16](#)
- [Adding Virtual Router Objects on page 17](#)

### Adding a Router Object

To add a router object:

1. Click **Configure**, expand **Shared**, and then click **Network**.  
The Shared Network pane appears.
2. From the Create new list, select **Device**.
3. Type a name for the new device in the dialog box, and click **OK**.  
The Device pane appears.
4. Enter information as described in the Help text in the main pane, and click **Apply**.

### Adding Virtual Router Objects

To add a virtual router to an existing router object:

1. Click **Configure**, expand **Shared>Network**, and then click an existing device.  
The Device pane appears.
2. From the Create new list, select **Virtual Router**.
3. Type a name for the new device in the dialog box, and click **OK**.  
The Virtual Router pane appears.
4. Click **Create**, enter information as described in the Help text in the main pane, and then click **Apply**.

#### Related Documentation

- [Using the NIC Resolver in Environments That Have Third-Party Devices \(C-Web Interface\) on page 28](#)
- [Configuration Tasks for Integrating Third-Party Network Devices \(C-Web Interface\) on page 12](#)
- [Adding Objects for Network Devices \(SRC CLI\) on page 14](#)
- [Integrating Network Devices into the SRC Network Overview on page 3](#)

### Setting Up SAE Communities (SRC CLI)

Tasks to configure SAE communities are:



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

1. [Configuring the SAE Community Manager on page 17](#)
2. [Specifying the Community Manager in the SAE Device Driver on page 18](#)

### Configuring the SAE Community Manager

Use the following configuration statements to configure the SAE community manager that manages third-party network 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, *sae\_mgr* is the name of the community manager.

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

## Specifying the Community Manager in the SAE Device Driver

Use the following configuration statements to specify the community manager in the SAE device driver.



```
shared sae configuration driver third-party {
  sae-community-manager sae-community-manager ;
}
```

To specify the community manager:

1. From configuration mode, access the configuration statements for the third-party device driver.

```
user@host# edit shared sae configuration driver third-party
```

2. Specify the name of the community manager.

```
[edit shared sae configuration driver third-party]
user@host# set sae-community-manager sae-community-manager
```

3. (Optional) Verify the configuration of the third-party device driver.

```
[edit shared sae configuration driver third-party]
user@host# show
sae-community-manager sae_mgr;
```

#### Related Documentation

- [Setting Up SAE Communities \(C-Web Interface\) on page 19](#)
- [Configuration Tasks for Integrating Third-Party Network Devices \(SRC CLI\) on page 11](#)
- [Configuring the SAE Community Manager](#)
- [Integrating Network Devices into the SRC Network Overview on page 3](#)
- [Logging In Subscribers and Creating Sessions on page 5](#)

## Setting Up SAE Communities (C-Web Interface)

Tasks to configure SAE communities are:

- [Configuring the SAE Community Manager on page 19](#)
- [Specifying the Community Manager in the SAE Device Driver on page 20](#)

### Configuring the SAE Community Manager

To configure the SAE community manager that manages third-party device communities:

1. Click **Configure**, expand **Shared>SAE**, and then click the SAE group for which you want to manage third-party devices.

The Group pane appears.

2. In the side pane, expand **Configuration>External Interface Features: PCMMCommunityManager**, and then click **Community Manager**.

The Community Manager pane appears.

3. Enter information as described in the Help text in the main pane, and click **Apply**.

## Specifying the Community Manager in the SAE Device Driver

To specify the community manager in the SAE device driver:

1. Click **Configure**, expand **Shared>SAE>Configuration>Driver**, and then click **Third Party**.

The Third Party pane appears.

2. Click **Create**, enter information as described in the Help text in the main pane, and then click **Apply**.

### Related Documentation

- [Setting Up SAE Communities \(SRC CLI\) on page 17](#)
- [Setting Up Script Services on page 13](#)
- [Adding Objects for Network Devices \(C-Web Interface\) on page 16](#)

---

## Configuring SAE Properties for the Event Notification API (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  
EventAPI]  
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
```

- (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 Documentation**
- [Using the SAE in a PCMM Environment](#)
  - [Configuring SAE Properties for the Event Notification API \(C-Web Interface\) on page 21](#)
  - [Initially Configuring the SAE](#)
  - [Configuring the SAE to Manage PCMM Devices \(SRC CLI\)](#)

## Configuring SAE Properties for the Event Notification API (C-Web Interface)

To configure properties for the event notification API:

- Click **Configure**, expand **Shared>SAE**, and then expand the SAE group for which you want to manage devices.
- In the side pane, expand **Configuration>External Interface Features: event**, and then click **Event API**.

The Event API pane appears.

- Enter information as described in the Help text in the main pane, and click **Apply**.

- Related Documentation**
- [Configuring an SAE Group](#)
  - [Configuring SAE Properties for the Event Notification API \(SRC CLI\) on page 20](#)

## Developing Router Initialization Scripts for Network Devices and Juniper Networks Routers

When the SAE establishes a connection with a router or network device, it can run an initialization script to customize the setup of the connection. These initialization scripts are run when the connection between a router or network device and the SAE is established and again when the connection is dropped.

We provide the `IorPublisher` script in the `/opt/UMC/sae/lib` folder. The `IorPublisher` script publishes the interoperable object reference (IOR) of the SAE in the directory so that a NIC can associate a router with an SAE.

For JunosE VRs that supply IP addresses from a local pool, a router initialization script is provided that identifies which VR supplies each IP pool and writes the information to the configuration. The SAE runs the script only when a COPS connection is established to the JunosE router. Consequently, if you modify information about IP pools on a VR after the COPS connection is established, the SAE will not automatically register the changes, and you must update the configuration.

For Junos (only junos-ise device) virtual routers that supply IP addresses from a local pool, a router initialization script is provided to get the IP pool information from Junos router and update it in LDAP.

Table 4 on page 22 describes the router initialization scripts that we provide with the SRC software in the `/opt/UMC/sae/lib` folder.

**Table 4: Router Initialization Scripts**

Script Name	Function	When to Use Script
iorPublisher	Publishes the IOR of the SAE into an internal part of the shared configuration so that a NIC can associate a router with an SAE.	Use with JunosE routers that do not supply IP addresses from local pools, and with devices running Junos OS.  Use with all devices running Junos OS.  Use with third-party network devices.
poolPublisher	Publishes the IOR of the SAE and local IP address pools in the directory so that a NIC can associate a router with an SAE and resolve the IP-to-SAE mapping.	Use with JunosE virtual routers that supply IP addresses from local pools.
isePoolPublisher	Publishes the local IP address pools in the directory so that a NIC can associate a router with an SAE and resolve the IP-to-SAE mapping.	Use with Junos (only junos-ise device) virtual routers that supply IP addresses from local pools.

## Interface Object Fields

Router initialization scripts are written in the Python programming language ([www.python.org](http://www.python.org)) and executed in the Jython environment ([www.jython.org](http://www.jython.org)).

Router initialization scripts interact with the SAE through an interface object called `Ssp`. The SAE exports a number of fields through the interface object to the script and expects the script to provide the entry point to the SAE.

Table 5 on page 23 describes the fields that the SAE exports.

Table 5: Exported Fields

Ssp Attribute	Description
Ssp.properties	System properties object (class: java.util.Properties)—The properties should be treated as read-only by the script.
Ssp.errorLog	Error logger—Use the SsperrorLog.println (message) to send error messages to the log.
Ssp.infoLog	Info logger—Use the Ssp.infoLog.println (message) to send informational messages to the log.
Ssp.debugLog	Debug logger—Use the Ssp.debugLog.println (message) to send debug messages to the log.

The router initialization script must set the field Ssp.routerInit to a factory function that instantiates a router initialization object:

- <VRName>—Name of the virtual router in which the COPS client has been configured, format: virtualRouterName@RouterName
- <virtualIp>—Virtual IP address of the SAE (string, dotted decimal; for example: 192.168.254.1)
- <realIp>—Real IP address of the SAE (string, dotted decimal; for example, 192.168.1.20)
- <VRip>—IP address of the virtual router (string, dotted decimal)
- <transportVR>—Name of the virtual router used for routing the COPS connection, or None, if the COPS client is directly connected

The factory function must implement the following interface:

```
Ssp.routerInit(VRName,
virtualIp,
realIp,
VRip,
transportVR)
```

The factory function returns an interface object that is used to set up and tear down a connection for a given COPS server. A common case of a factory function is the constructor of a class.

The factory function is called directly after a COPS server connection is established. In case of problems, an exception should be raised that leads to the termination of the COPS connection.

## Required Methods

Instances of the interface object must implement the following methods:

- *setup()*—Is called when the COPS server connection is established and is operational. In case of problems, an exception should be raised that leads to the termination of the COPS connection.

- `shutdown()`—Is called when the COPS server connection to the virtual router is terminated. This method should not raise any exceptions in case of problems.

## Example: Router Initialization Script

The following script defines a router initialization class named `SillyRouterInit`. The interface class does not implement any useful functionality. The interface class just writes messages to the infoLog when the router connection is created or terminated.

```
class SillyRouterInit:
    def __init__(self, vrName, virtualIp, realIp, vrIp, transportVr):
        """ initialize router initialization object """
        self.vrName = vrName
        Ssp.infoLog.println("SillyRouterInit created")

    def setup(self):
        """ initialize connection to router """
        Ssp.infoLog.println("Setup connection to VR %(vrName)s" %
            vars(self))

    def shutdown(self):
        """ shutdown connection to router """
        Ssp.infoLog.println("Shutdown connection to VR %(vrName)s" %
            vars(self))

#
# publish interface object to Ssp core
#
Ssp.routerInit = SillyRouterInit
```

### Related Documentation

- [How SNMP Obtains Information from Routers for the SRC Software](#)
- [Specifying JunosE Router Initialization Scripts on the SAE \(SRC CLI\)](#)
- [Accessing the Router CLI](#)
- [Viewing Statistics for Specific JunosE Device Drivers \(SRC CLI\)](#)
- [Troubleshooting Problems with Managing JunosE Routers](#)
- [Updating Local IP Address Pools for Junos Virtual Routers \(SRC CLI\)](#)
- [Updating Local IP Address Pools for JunosE Virtual Routers \(SRC CLI\)](#)

## Copying Initialization Scripts to the C Series Controller

---

If you use a script that is not provided with the SRC module, you need to use the `file copy` command to copy your script to the C Series Controller. For example:

```
user@host> file copy ftp://user@myserver/routerinit.py /opt/UMC/sae/lib
Password:
```

### Related Documentation

- [Specifying Initialization Scripts on the SAE \(SRC CLI\) on page 25](#)
- [Setting Up Script Services on page 13](#)
- [Developing Router Initialization Scripts for Network Devices and Juniper Networks Routers on page 21](#)

## Specifying Initialization Scripts on the SAE (SRC CLI)

Use the following configuration statements to specify initialization scripts for third-party devices:

```
shared sae configuration driver scripts {
  extension-path extension-path ;
  general general ;
}
```

To configure initialization scripts for third-party devices:

1. From configuration mode, access the configuration statements that configure initialization scripts.

```
user@host# edit shared sae configuration driver scripts
```

2. Specify the initialization script for third-party devices.

```
[edit shared sae configuration driver scripts]
user@host# set general general
```

3. Configure a path to scripts that are not in the default location, `/opt/UMC/sae/lib`.

```
[edit shared sae configuration driver scripts]
user@host# set extension-path extension-path
```

4. (Optional) Verify your initialization script configuration.

```
[edit shared sae configuration driver scripts]
user@host# show
```

### Related Documentation

- [Specifying Initialization Scripts on the SAE \(C-Web Interface\) on page 25](#)
- [Copying Initialization Scripts to the C Series Controller on page 24](#)
- [Developing Router Initialization Scripts for Network Devices and Juniper Networks Routers on page 21](#)

## Specifying Initialization Scripts on the SAE (C-Web Interface)

To configure initialization scripts for third-party devices:

1. Click **Configure**, expand **Shared>SAE>Configuration>Driver**, and then click **Scripts**.  
The Scripts pane appears.
2. Click **Create**, enter information as described in the Help text in the main pane, and then click **Apply**.

### Related Documentation

- [Specifying Initialization Scripts on the SAE \(SRC CLI\) on page 25](#)
- [Copying Initialization Scripts to the C Series Controller on page 24](#)

- [Developing Router Initialization Scripts for Network Devices and Juniper Networks Routers on page 21](#)

## Using SNMP to Retrieve Information from Network Devices

---

You can use SNMP to retrieve information from a network device. For example, if you create a script that uses SNMP, specify the SNMP communities that are on the network device.

To retrieve information:

- (Recommended) Specify SNMP communities for each virtual router object.
- Configure global default SNMP communities.

### Related Documentation

- [Adding Virtual Router Objects \(SRC CLI\) on page 15](#)
- [Adding Objects for Network Devices \(C-Web Interface\) on page 16](#)
- [Configuring Global SNMP Communities in the SRC Software \(SRC CLI\) on page 26](#)
- [Configuring Global SNMP Communities in the SRC Software \(C-Web Interface\) on page 27](#)

## Configuring Global SNMP Communities in the SRC Software (SRC CLI)

---

You can configure global default SNMP communities that are used if a VR does not exist on the router or if the community strings have not been configured for the VR.

Use the following configuration statements to configure global default SNMP communities:

```
shared sae configuration driver snmp {  
  read-only-community-string read-only-community-string;  
  read-write-community-string read-write-community-string;  
}
```

To configure global default SNMP communities:

1. From configuration mode, access the statements that configure default SNMP communities.

```
user@host# edit shared sae configuration driver snmp
```

2. Configure the default SNMP community string used for read access to the router.

```
[edit shared sae configuration driver snmp]  
user@host# set read-only-community-string read-only-community-string
```

3. Configure the default SNMP community string used for write access to the router.

```
[edit shared sae configuration driver snmp]  
user@host# set read-write-community-string read-write-community-string
```



- (Optional) Verify your configuration.

```
[edit shared sae configuration driver snmp]
user@host# show
read-only-community-string *****;
read-write-community-string *****;
```

**Related Documentation**

- [Using SNMP to Retrieve Information from JunosE Routers and Devices Running Junos OSs \(SRC CLI\)](#)
- [Configuring Global SNMP Communities in the SRC Software \(C-Web Interface\) on page 27](#)
- [Using SNMP to Retrieve Information from Network Devices on page 26](#)
- [How SNMP Obtains Information from Routers for the SRC Software](#)

## Configuring Global SNMP Communities in the SRC Software (C-Web Interface)

You can configure global default SNMP communities that are used if a VR does not exist on the router or if the community strings have not been configured for the VR.

To configure global default SNMP communities:

- Click **Configure**, expand **Shared>SAE>Configuration>Driver**, and then click **SNMP**.  
The SNMP pane appears.
- Click **Create**, enter information as described in the Help text in the main pane, and then click **Apply**.

**Related Documentation**

- [Configuring Global SNMP Communities in the SRC Software \(SRC CLI\) on page 26](#)

## Using the NIC Resolver in Environments That Have Third-Party Devices (SRC CLI)

If you are using the assigned IP subscriber method of logging in subscribers, and you are using the NIC to determine the subscriber's SAE, you need to configure a resolver on the NIC. The OnePopDynamicIp sample configuration data supports this scenario. The OnePopDynamicIp configuration supports one point of presence (POP) and provides no redundancy. The realm for this configuration accommodates the situation in which IP pools are configured locally on each virtual router object.

You can access the OnePopDynamicIp configuration in the SRC CLI.

**Related Documentation**

- [Configuration Tasks for Integrating Third-Party Network Devices \(SRC CLI\) on page 11](#)
- [Integrating Network Devices into the SRC Network Overview on page 3](#)
- [Configuring the NIC \(SRC CLI\)](#)

## Using the NIC Resolver in Environments That Have Third-Party Devices (C-Web Interface)

---

If you are using the assigned IP subscriber method of logging in subscribers, and you are using the NIC to determine the subscriber's SAE, you need to configure a resolver on the NIC. The OnePopDynamicIp sample configuration data supports this scenario. The OnePopDynamicIp configuration supports one point of presence (POP) and provides no redundancy. The realm for this configuration accommodates the situation in which IP pools are configured locally on each virtual router object.

You can access the OnePopDynamicIp configuration.

### **Related Documentation**

- [Locating Subscriber Management Information](#)
- [Integrating Network Devices into the SRC Network Overview on page 3](#)
- [Configuration Tasks for Integrating Third-Party Network Devices \(C-Web Interface\) on page 12](#)

## PART 3

# Administration

- [Monitoring Third-Party Network Devices with Scripting Service on page 31](#)



## CHAPTER 4

# Monitoring Third-Party Network Devices with Scripting Service

- Viewing Information About SAE Device Drivers (SRC CLI) on page 31
- Viewing Information About Device Drivers (C-Web Interface) on page 32
- Viewing SNMP Information for Routers and Devices (SRC CLI) on page 33
- Viewing Statistics for Device Drivers (SRC CLI) on page 34
- Viewing SNMP Statistics for Devices (C-Web Interface) on page 35
- Viewing Statistics for Specific Device Drivers (SRC CLI) on page 36
- Viewing SNMP Statistics for Specific Devices (C-Web Interface) on page 37

### Viewing Information About SAE Device Drivers (SRC CLI)

**Purpose** View information about SAE device drivers. Each device driver manages one logical router instance.

**Action** To view information about the state of SAE device drivers:

```
user@host> show sae drivers
JunosE Driver
Device name          default@dryad
Device type          junose
Device IP            10.227.7.244
Local IP             10.227.7.172
TransportRouter      default@dryad
Device version       7.2.0
Start time           Tue Feb 13 14:18:44 EST 2007
Number of notifications 20
Number of processed added 14
Number of processed changed 0
Number of processed deleted 6
Number of provisioning attempt 30
Number of provisioning attempt failed 0
Number of outstanding decisions 0
Number of SAP        7
Number of PAP        1
Number of active user sessions 100
Number of active service sessions 0
  Job Queue
  Size              0
  Age (ms)          1
```

```

Total enqueued          28
Total dequeued          28
Average job time (ms) 426
State Synchronization
Number recovered subscriber sessions      0
Number recovered service sessions        0
Number recovered interface sessions      0
Number invalid subscriber sessions       0
Number invalid service sessions         0
Number invalid interface sessions        0
Background restoration start time        Tue Feb 13 14:18:49 EST 2007

Background restoration end time          Tue Feb 13 14:18:49 EST 2007

Number subscriber sessions restored in background 0
Number of provisioning objects left to collect  0
Total number of provisioning objects to collect 11
Start time                                     Tue Feb 13 14:18:45 EST 2007

End time                                       Tue Feb 13 14:18:47 EST 2007

Number of synched contexts                 7
Number of post-sync jobs                  6
Session Store Info
Session Store Status                      sessionsCollected
Status Last Update Time                  Mon Jul 29 10:26:26 UTC 2013
Current Usage Ratio                      1.0
Last Modified Time                      Size(KB) Name LiveSessionsSize(KB)
Mon Jul 29 10:27:05 UTC 2013  117.9      storeOps_1_1 117.9

```

To view information about the state of a particular device driver, specify all or part of the virtual router name. For device drivers running Junos OS and PCMM drivers, use the format `default@routerName`.

```
user@host> show sae drivers device-name device-name
```

To view only the virtual router names for the device driver:

```
user@host> show sae drivers brief
```

```

Router Drivers
Router Name      Router Type
default@simJunos  junos

```

To restrict the number of displayed results:

```
user@host> show sae drivers maximum-results maximum-results
```

#### Related Documentation

- [Initially Configuring the SAE](#)
- [Shutting Down the Device Drivers \(SRC CLI\)](#)
- [Viewing Information About Device Drivers \(C-Web Interface\) on page 32](#)
- [Viewing Statistics for Device Drivers \(SRC CLI\) on page 34](#)

## Viewing Information About Device Drivers (C-Web Interface)

**Purpose** View information about the device drivers available on the SAE.

- Action** 1. Click **Monitor>SAE >Drivers**.

The Drivers pane appears.

2. In the Device Name box, enter a full or partial device driver name for which you want to display information, or leave the box blank to display all devices.

For JunosE router drivers, use the format:

**<virtual router name>@<router name>**

For device drivers running Junos OS and PCMM drivers, use the format:

**default@<router name>**

3. In the Maximum Results box, enter the maximum number of results that you want to receive.
4. In the Slot box, enter the number of the slot for which you want to display device information.
5. Select an output style from the Style list.
6. Click **OK**.

The Drivers pane displays the status of the devices running on the SAE.

**Related Documentation**

- [Connections to Managed Devices](#)
- [Viewing SNMP Information for Routers and Devices \(SRC CLI\) on page 33](#)
- [Viewing Statistics for Device Drivers \(SRC CLI\) on page 34](#)
- [Viewing Statistics for Specific Device Drivers \(SRC CLI\) on page 36](#)
- [Viewing Information About SAE Device Drivers \(SRC CLI\) on page 31](#)

## Viewing SNMP Information for Routers and Devices (SRC CLI)

- Purpose** View SNMP information for routers and devices that the SAE manages. You can view information for all routers and devices, or for specific ones.

**Action** To view SNMP information for routers and devices that the SAE is managing:

```
user@host> show sae statistics device
```

To view information for a particular router, specify all or part of the VR name. For device drivers running Junos OS and PCMM drivers, use the format default@routerName.

```
user@host> show sae statistics device filter filter
```

To view only the RADIUS clients that were accessible by IP address and port number:

```
user@host> show sae statistics device brief
```

- Related Documentation**
- [Configuring SAE Properties for Global Default SNMP Communities for Use with JunosE Routers and Devices Running Junos OS](#)
  - [Viewing Statistics for Device Drivers \(SRC CLI\) on page 34](#)
  - [Viewing SNMP Information for Licenses on Virtual Routers \(SRC CLI\)](#)
  - [Viewing Statistics for Specific Device Drivers \(SRC CLI\) on page 36](#)

---

## Viewing Statistics for Device Drivers (SRC CLI)

---

**Purpose** View SNMP statistics for all device drivers.



**Action** user@host> show sae statistics device common

**SNMP Statistics**

```
Driver type           JunosE COPS
Number of close requests 0
Number of connections accepted 0
Number of current connections 0
Number of open requests 0
Server address        0.0.0.0
Server port           3288
Time since last redirect 0
```

**SNMP Statistics**

```
Driver type           PACKETCABLE COPS
Number of close requests 0
Number of connections accepted 0
Number of current connections 0
Number of open requests 0
Server address        0.0.0.0
Server port           0
Time since last redirect 0
```

**SNMP Statistics**

```
Driver type           Junos
Number of close requests 0
Number of connections accepted 0
Number of current connections 0
Number of open requests 0
Server address        0.0.0.0
Server port           3333
Time since last redirect 0
```

The value of the server address can be either an IPv4 or IPv6 address, depending on the platform.

- Related Documentation**
- [Shutting Down the Device Drivers \(C-Web Interface\)](#)
  - [Viewing Information About SAE Device Drivers \(SRC CLI\) on page 31](#)
  - [Viewing SNMP Information for Routers and Devices \(SRC CLI\) on page 33](#)
  - [Viewing Statistics for Specific Device Drivers \(SRC CLI\) on page 36](#)

## Viewing SNMP Statistics for Devices (C-Web Interface)

---

**Purpose** View SNMP statistics about devices.

**Action** 1. Click **Monitor>SAE >Statistics>Device**.

The Statistics/Device pane appears.

The screenshot shows the Juniper C-Web Interface with the SAE Statistics / Device configuration pane. The pane has a left sidebar with a menu including ACP, CLI, Component, Date, Disk, Interfaces..., Iptables..., JPS, IHC, ITP, Redirect Server, Route..., SAE, Security, and System. The main area contains the following fields and options:

- Device Name:** A text input field. Help text: "Name of a device. Value: All or part of the device name. For JUNOS router drivers, use the format virtualRouterName@routerName. For JUNOS router drivers and PCMM drivers, use the format default@routerName. Default: No value".
- Slot:** A text input field. Help text: "Display SAE information for a specified slot. Value: Currently the chassis has only one slot. The valid value is 0. Default: 0".
- Style:** A dropdown menu. Help text: "Output style Choices: brief: Display only device names Default: Detail".

At the bottom of the pane are "OK" and "Reset" buttons. The footer of the interface shows "Copyright © 2007, Juniper Networks, Inc. All Rights Reserved. Trademark Notice. Privacy." and the Juniper logo.

- In the Device Name box, enter a full or partial device name for which you want to display information, or leave the box blank to display all devices.
- In the Slot box, enter the number of the slot for which you want to display SNMP statistics for devices.
- Select an output style from the Style list.
- Click **OK**.

The Statistics/Device pane displays statistics for all devices.

#### Related Documentation

- [Configuring SAE Properties for Global Default SNMP Communities for Use with JunosE Routers and Devices Running Junos OS](#)
- [Viewing SNMP Statistics for Specific Devices \(C-Web Interface\) on page 37](#)
- [Viewing SNMP Statistics for Subscriber Sessions and Service Sessions \(C-Web Interface\)](#)

## Viewing Statistics for Specific Device Drivers (SRC CLI)

**Purpose** View statistics for specific router drivers or device drivers.

**Action** To view SNMP statistics for device drivers running Junos OS:

```
user@host> show sae statistics device common junos
```

To view SNMP statistics for JunosE router drivers:

```
user@host> show sae statistics device common junose-cops
```

To view SNMP statistics for PCMM device drivers:

```
user@host> show sae statistics device common packetcable-cops
```

To view SNMP statistics for third-party device drivers:

```
user@host> show sae statistics device common proxy
```

For example, to view SNMP statistics for device drivers running Junos OS:

```
user@host> show sae statistics device common junos
```

```
SNMP Statistics
Driver type           Junos OS
Number of close requests 0
Number of connections accepted 0
Number of current connections 0
Number of open requests 0
Server address        0.0.0.0
Server port           3333
Time since last redirect 0
```

#### Related Documentation

- [Configuring the Session Store Feature \(SRC CLI\)](#)
- [Viewing Information About SAE Device Drivers \(SRC CLI\) on page 31](#)
- [Viewing SNMP Information for Routers and Devices \(SRC CLI\) on page 33](#)
- [Viewing Statistics for Device Drivers \(SRC CLI\) on page 34](#)

## Viewing SNMP Statistics for Specific Devices (C-Web Interface)

**Purpose** View SNMP statistics about specific devices.

**Action** 1. Click **Monitor>SAE >Statistics>Device>Common**.

The Statistics/Device/Common pane appears.

The screenshot shows the Juniper C-Web Interface with the SAE Statistics / Device / Common configuration pane. The pane has a left sidebar with navigation options: ACP, CLI, Component, Date, Disk, Interfaces..., Iptables..., JPS, IIC, ITP, Redirect Server, Route..., SAE, Security, and System. The main content area is titled 'Statistics / Device / Common' and contains three input fields: 'Device Name', 'Slot', and 'Type'. Each field has a description and a default value. The 'Device Name' field has a description: 'Name of a device. Value: All or part of the device name.' and two bullet points: 'For JUNOSe router drivers, use the format virtualRouterName@routerName.' and 'For JUNOS router drivers and PCMM drivers, use the format default@routerName.'. The 'Slot' field has a description: 'Display SAE information for a specified slot. Value: Currently the chassis has only one slot. The valid value is 0.' and a default value of '0'. The 'Type' field has a description: 'Display SNMP statistics for a specified device driver type.' and a list of choices: 'junos: Display SNMP statistics for JUNOS router drivers', 'junose-cops: Display SNMP statistics for JUNOSe router drivers', 'packetable-cops: Display SNMP statistics for PCMM device drivers', and 'proxy: Display SNMP statistics for third-party drivers'. The default value for 'Type' is 'No value'. At the bottom of the pane are 'OK' and 'Reset' buttons. The footer of the interface shows 'Copyright © 2007, Juniper Networks, Inc. All Rights Reserved. Trademark Notice. Privacy.' and the Juniper logo.

2. In the Device Name box, enter a full or partial device name for which you want to display information, or leave the box blank to display all devices.
3. In the Slot box, enter the number of the slot for which you want to display SNMP statistics for specific devices.
4. Select a device type from the Type list:
  - junos—Displays SNMP statistics for device drivers running Junos OS
  - junose-cops—Displays SNMP statistics for JunosE router drivers

- packetcable-COPS—Displays SNMP statistics for PCMM device drivers
- proxy—Displays SNMP statistics for third-party drivers

5. Click **OK**.

The Statistics/Device/Common pane displays statistics for the specified device.

**Related  
Documentation**

- *Configuring SAE Properties for Global Default SNMP Communities for Use with JunosE Routers and Devices Running Junos OS*
- [Viewing SNMP Statistics for Devices \(C-Web Interface\) on page 35](#)
- *Viewing SNMP Statistics for Subscriber Sessions and Service Sessions (C-Web Interface)*

PART 4

# Troubleshooting

- [Troubleshooting Procedures on page 41](#)



# Troubleshooting Procedures

- [Collecting Data with the Activity Monitor \(SRC CLI\) on page 41](#)
- [Collecting Data with the Activity Monitor \(C-Web Interface\) on page 42](#)

## Collecting Data with the Activity Monitor (SRC CLI)

---

You can collect data with the Activity Monitor for specific components over a specified time and save them to a tar.gz file in the `/opt/UMC/activity/var/agnostic/*` directory. You can view the exact file name and path after you execute the **request support information** command. Before you perform data collection with the Activity Monitor, make sure the filter for the specific components is enabled.

To perform data collection with the Activity Monitor:

- `user@host> request support information`  
Some of the information retrieved includes:
  - System log messages from the `/var/log/messages/*` directory.
  - The configuration in text format, XML format, and set format.
  - The host name in the name of the diagnostic file.

To perform data collection for specific components:

- `user@host> request support information component`  
where *component* is one of the following:
  - `acp`—SRC Admission Control Plug-In
  - `activity`—Activity Monitor
  - `agent`—SNMP agent
  - `appsvr`—Application server
  - `cli`—SRC CLI
  - `diameter`—Diameter application
  - `dsa`—Dynamic Service Activator
  - `extsubmon`—External Subscriber Monitor

- ims—IP multimedia subsystem
- jdb—Juniper Networks database
- jps—Juniper Policy Server
- licSvr—License server
- nic—Network information collector
- redir—Redirect server
- sae—SAE
- webadm—C-Web interface

To perform data collection for a specified number of days:

- user@host> **request support information days**  
where **days** is in the range of 1–36500.

**Related  
Documentation**

- *Before You Load a Configuration*
- *Viewing Graphs (C-Web Interface)*
- *Viewing Graphs from a Web Page*
- *Monitoring Activity on C Series Controllers*

## Collecting Data with the Activity Monitor (C-Web Interface)

---

You can collect data with the Activity Monitor for specific components over a specified time. Before you configure data collection for the Activity Monitor, make sure the Activity Monitor (activity), CLI (cli), and C-Web interface (webadm) components are enabled.

To perform data collection with the Activity Monitor:

1. Click **Manage>Request>Support>Information**.  
The Support Information pane appears.
2. From the Components list, select the components you want to monitor, and click **OK**.
3. (Optional) Enter the number of days for which you want to collect data, and click **OK**.

**Related  
Documentation**

- *Viewing Graphs (C-Web Interface)*
- *Viewing Graphs from a Web Page*
- *Monitoring Activity on C Series Controllers*



PART 5

# Index

- [Index on page 45](#)



# Index

## A

Activity Monitor	
data collection.....	41, 42
assigned IP subscribers	
third-party devices.....	5
IP address pools.....	5

## C

community manager	
configuring, third-party devices	
C-Web interface.....	19
SRC CLI.....	17
conventions	
notice icons.....	x
text.....	x
customer support.....	xii
contacting JTAC.....	xii

## D

device drivers	
viewing on SAE	
C-Web interface.....	32
SRC CLI.....	31
documentation	
comments on.....	xii

## E

event notification, PCMM network	
configuration statements.....	20
properties, configuring	
C-Web interface.....	21
SRC CLI.....	20
event notification, third-party devices	
description.....	6
properties, configuring	
C-Web interface.....	21

## I

IOR	
router initialization scripts.....	22

## L

login process	
assigned IP subscribers, third-party devices.....	5
event notification method, third-party devices.....	7

## M

manuals	
comments on.....	xii

## N

network devices	
SNMP communities, configuring	
C-Web interface.....	27
SRC CLI.....	26
SNMP information, viewing	
C-Web interface.....	35, 37
notice icons.....	x

## R

router initialization scripts	
developing.....	21
iorPublisher.....	22
isePoolPublisher.....	22
JunosE	
example.....	24
JunosE Software.....	21
poolPublisher.....	22
router object	
adding for third-party devices	
SRC CLI.....	14
routers	
SNMP communities, configuring	
C-Web interface.....	27
SRC CLI.....	26

## S

SAE (service activation engine)	
router initialization scripts. <i>See</i> router initialization scripts	
SAE (service activation engine), configuring	
event notification API properties	
C-Web interface.....	21
SRC CLI.....	20
SAE communities	
configuring, third-party devices	
SRC CLI.....	17
description, third-party devices.....	3

script services	
for third-party devices.....	4
session store	
in third-party networks.....	4
SNMP	
retrieving information from network	
devices.....	26
SNMP agent	
viewing information on SAE	
C-Web interface.....	35, 37
SNMP communities	
configuring	
C-Web interface.....	27
SRC CLI.....	26
support, technical See technical support	

## T

technical support	
contacting JTAC.....	xii
text conventions.....	x
third-party devices	
creating sessions.....	5
integrating into SRC network	
SRC CLI.....	3
logging in subscribers	
assigned IP method.....	5
overview.....	5
provisioning with script services.....	4
router objects, adding	
SRC CLI.....	14
SAE communities.....	3
VR objects, adding	
C-Web interface.....	17
SRC CLI.....	15

## V

virtual routers	
adding for third-party devices	
C-Web interface.....	17
SRC CLI.....	15