Chapter 18

Using JUNOS Routing Platforms in the SRC Network with the C-Web Interface

This chapter describes how to use the C-Web interface to set up the SRC software and how to set up JUNOS routing platforms so that the routing platforms can be used in the SRC network. It also shows how to monitor the interactions between the SAE and JUNOS routing platforms and how to troubleshoot SRC problems on JUNOS routing platforms.

You can also use the following to configure JUNOS routers:

- To use the SRC CLI, see *SRC-PE Network Guide, Chapter 7, Using JUNOS Routing Platforms in the SRC Network with the SRC CLI*.
- To use the Solaris platform, see *SRC-PE Network Guide, Chapter 8, Using JUNOS Routing Platforms in the SRC Network with a Solaris Platform*.

Topics in this chapter include:

- BEEP Connection Between JUNOS Routing Platforms and the SAE on page 170
- Adding JUNOS Routing Platforms and Virtual Routers with the C-Web Interface on page 170
- Configuring the SAE to Manage JUNOS Routing Platforms with the C-Web Interface on page 172
- Configuring Secure Connections Between the SAE and JUNOS Routing Platforms on page 172
- Checking Changes to the JUNOS Configuration with the C-Web Interface on page 176
- Using SNMP to Retrieve Information from JUNOS Routing Platforms on page 177
- Developing Router Initialization Scripts on page 177
- Specifying JUNOS Router Initialization Scripts on the SAE with the C-Web Interface on page 179
- Accessing the Router CLI on page 180

- Configuring JUNOS Routing Platforms to Interact with the SAE on page 180
- Disabling Interactions Between the SAE and JUNOS Routing Platforms on page 181
- Monitoring Interactions Between the SAE and JUNOS Routing Platforms on page 182
- Troubleshooting Problems with the SRC Software Process with the C-Web Interface on page 182

BEEP Connection Between JUNOS Routing Platforms and the SAE

For information about which JUNOS routing platforms and releases a particular SRC release supports, see the SRC *Release Notes*.

The SAE interacts with a JUNOS software process, referred to as the SRC software process in this documentation, on the JUNOS routing platform. The SAE and the SRC software process communicate using the Blocks Extensible Exchange Protocol (BEEP). You can secure the BEEP connection by using Transport Layer Security (TLS).

When the SRC software process establishes a BEEP session for the SAE, the SAE configures an interface on the JUNOS routing platform. The SAE builds the configuration for an interface using the policies stored in the directory. If the policies are subsequently modified, the SAE builds a new configuration and reconfigures the interface on the JUNOS routing platform.The JUNOS routing platform stores data about interfaces and services that the SAE manages in a configuration group called sdx. You must create this configuration group on the JUNOS routing platform.

Adding JUNOS Routing Platforms and Virtual Routers with the C-Web Interface

On JUNOS routing platforms, the SAE manages interfaces. The SRC software associates a virtual router called default with each JUNOS routing platform. Each JUNOS routing platform in the SRC network and its associated virtual router (VR) called default must appear in the directory. The VRs are not actually configured on the JUNOS routing platform; the VR in the directory provides a way for the SAE to manage the interfaces on the JUNOS routing platform.

There are two ways to add routers:

- Detect operative routers and configured JUNOS VRs in the SRC network and add them to the configuration.
- Add each router and VR individually.

Adding Operative JUNOS Routing Platforms

To add to the directory routers and JUNOS VRs that are currently operative and have an operating SNMP agent:

1. Click **Manage > Request > Network > Discovery**.
2. Enter information as described in the Help text in the main pane, and click **Apply**.

If you add a router using the discover network feature, the software adds the IP address of the first SNMP agent on the router to respond to the discover request.

Adding JUNOS Routers Individually

To add a router:

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. From the Device Type list, select **JUNOS**.
5. Enter information as described in the Help text in the main pane, and click **Apply**.

Adding Virtual Routers Individually

To add a virtual router to an existing router:

1. Click **Configure**, expand **Shared > Network**, and then click a JUNOS router.

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

Related Topics

- For information about service scopes, see *SRC-PE Services and Policies Guide, Chapter 1, Managing Services with the SRC CLI*
- For information about tracking plug-ins, see *Chapter 27, Configuring Internal, External, and Synchronization Plug-Ins with the C-Web Interface*

Configuring the SAE to Manage JUNOS Routing Platforms with the C-Web Interface

A JUNOS routing platform interacts with the SAE by using a JUNOS software process called sdxd. When the sdxd process establishes a TCP/IP connection to the SAE, the SAE begins to manage the router. The JUNOS router driver configuration defines parameters related to the interactions between the SAE and the sdxd process.

To configure the SAE to manage JUNOSe routers:

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

   The JUNOS pane appears.

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

Related Topics

- For information about setting up SAE groups, see *Chapter 8, Setting Up an SAE with the C-Web Interface*.

Configuring Secure Connections Between the SAE and JUNOS Routing Platforms

You can use TLS to protect communication between the SAE and JUNOS routing platforms.

To complete the handshaking protocol for the TLS connection, the client (JUNOS routing platform) and the server (SAE) must exchange and verify certificates. You need to create a client certificate and a server certificate. Both certificates must be signed by a certificate authority (CA). JUNOS software supports VeriSign, Inc. (http://www.verisign.com). You must then install both certificates on the SAE and on the JUNOS routing platform.

You can use the C-Web interface to manage certificates manually, or through the Simple Certificate Enrollment Protocol (SCEP).

Certificates are in the format defined in the X.509 standard for public key infrastructure. The certificate requests are in the Public Key Cryptology Standard (PKCS) #10 format.

Tasks to set up the SAE and the JUNOS routing platform to use TLS are:

1. Manually Obtaining Digital Certificates on page 173

   Or

2. Obtaining Digital Certificates through SCEP on page 174
3. Installing the Server Certificate on the Router on page 174
4. Creating a Client Certificate for the Router on page 175
5. Installing the Client Certificate on the Router on page 175
6. Configuring the SAE to Use TLS on page 175
7. Configuring TLS on the SAE on page 176

Manually Obtaining Digital Certificates

You can manually add digital certificates, or you can use SCEP to help manage how you obtain certificates. See *Obtaining Digital Certificates through SCEP* on page 174.

To manually add a signed certificate:

1. Create a certificate signing request.

   a. Click **Manage > Request > Security > General Certificate Certificate**.

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

   By default, this request creates the file /tmp/certreq.csr and encodes the file by using Privacy-Enhanced Mail (pem) encoding.

2. Copy the file generated in Step 1 to another system, and submit the certificate signing request file generated in Step 1 to VeriSign, Inc. (http://www.verisign.com) for signing.

   You can transfer the file through FTP by using the file copy command.

   user@host> **file copy** *source_file* **ftp://***username***@***server*[:*port*]*/destination_file*

   VeriSign authenticates you and returns a certificate, signed by them, that authenticates your public key.

3. When you receive the signed certificate, copy the file back to the SRC system to the */tmp* directory.

   You can transfer the file through FTP, as shown in Step 2.

4. Add the certificate to the SRC configuration.

   a. Click **Manage > Request > Security > Import Certificate**.

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

Obtaining Digital Certificates through SCEP

You can use SCEP to help manage how you obtain digital certificates, or you can manually add certificates. See *Manually Obtaining Digital Certificates* on page 173.

Before you can obtain certificates for your use, you must get the CA's certificate and install it in the local store of trusted certificates.

To add a signed certificate that you obtain through SCEP:

1. Request your CA's certificate through SCEP.

   a. Click **Manage > Request > Security > Get GA Certificate**.

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

2. Request that the certificate authority automatically sign the certificate request:

   a. Click **Manage > Request > Security > Enroll**.

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

Installing the Server Certificate on the Router

The TLS client (JUNOS routing platform) needs a copy of the certificate that was used to sign the SAE certificate so that it can verify the SAE certificate. To install the SAE certificate on the JUNOS routing platform:

1. Include the following statements at the [edit security certificates certificate-authority] hierarchy level.

```
[edit security certificates certificate-authority]
security{
    certificates{
        certificate-authority SAECert{
            file /var/db/certs/cert.pem;
        }
    }
}
```

2. Include the following statements at the [system services service-deployment] hierarchy level.

```
system{
    services{
        service-deployment{
```

```
            servers {
               server-address port port-number{
                  security-options {
                     tls;
                  }
               }
            }
         }
      }
}
```

Creating a Client Certificate for the Router

For information about how to obtain a certificate for the router from a certificate authority, see *Obtaining a Certificate from a Certificate Authority* in the *JUNOS System Basics Configuration Guide*.

Installing the Client Certificate on the Router

To install the client (router) certificate on the JUNOS routing platform:

1. Include the following statements at the [edit security certificates certificate-authority] hierarchy level.

   ```
   [edit security certificates certificate-authority]
   security{
       certificates{
           local clientCERT { .... } ;
       }
   }
   ```

2. Include the following statements at the [system services service-deployment] hierarchy level.

   ```
   system{
       services{
           service-deployment{
               local-certificate clientCert;
           }
       }
   }
   ```

Configuring the SAE to Use TLS

To configure the SAE to accept TLS connections:

1. Click **Configure**, expand **Shared > Network**, and then click a JUNOS router.

   The Device pane appears.

2. Type a port number in the Beep Server Port box, and click **Apply**.

Configuring TLS on the SAE

To configure TLS on the SAE:

1. Click **Configure**, expand **Shared > SAE > Configuration > Driver > JUNOS**, and then click **Security**.

   The Security pane appears.

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

Checking Changes to the JUNOS Configuration with the C-Web Interface

The SAE can check the configuration of a JUNOS routing platform under its control to detect whether the configuration has changed by a means other than through the SAE. If the SAE finds a disparity between the router and the SAE configurations, it can take several actions. The SAE checks the configuration installed on the router against the state of the SAE session layer (subscriber, service, and interface sessions). While the check is occurring, the SAE does not handle jobs from the router, and all provisioning activity is blocked, including event notifications.

The SAE can take the following actions if it finds a disparity between the router and SAE configurations:

- The SAE takes the state of the session layer on the router to be correct and updates its local state to be consistent with the router. The SAE then sends stop events for all sessions where the corresponding provisioning in the router has been removed.

- The SAE takes its local state to be the correct state and updates the router to be consistent with its local state.

- The SAE does not solve the state discrepancy. It reports disparities through the SAE device driver event trap called routerConfOutOfSynch and through the info log.

Note that it is not possible to check the consistency of individual objects that the SAE provisions Therefore, modifications to a provisioning object while the SAE is disconnected from the router cannot be detected.

Setting Up Periodic Configuration Checking

Use the following configuration statements to configure the SAE to periodically check the configuration of the JUNOS routing platform:

To configure the SAE to periodically check the configuration of the JUNOS routing platform:

1. Click **Configure**, expand **Shared > SAE > Configuration > Driver > JUNOS**, and then click **Configuration Checking**.

   The Configuration Checking pane appears.

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

Using SNMP to Retrieve Information from JUNOS Routing Platforms

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

We recommend that you specify SNMP communities for each virtual router. (See *Adding Virtual Routers Individually* on page 171.) You can also configure global default SNMP communities.

Configuring Global SNMP Communities in the SRC Software

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:

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

   The SNMP pane appears.

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

Developing Router Initialization Scripts

When the SAE establishes a connection with a router, it can run a router initialization script to customize the setup of the connection. Router initialization scripts are run when the connection between a router 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 IOR of the SAE into an internal part of the shared configuration so that a NIC can associate a router with an SAE.

Interface Object Fields

Router initialization scripts are written in the Python programming language (www.python.org) and executed in the Jython environment (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 9 describes the fields that the SAE exports.

Table 9: 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.printIn (message) to send error messages to the log.
Ssp.infoLog	Info logger—Use the Ssp.infoLog.printIn (message) to send informational messages to the log.
Ssp.debugLog	Debug logger—Use the Ssp.debugLog.printIn (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, in the 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 is terminated to the virtual router. 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; it 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.printin("SillyRouterInit created")

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

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

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

Specifying JUNOS Router Initialization Scripts on the SAE with the C-Web Interface

To configure router initialization scripts for JUNOS routing platforms:

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**.

Accessing the Router CLI

You can access the CLIs of Juniper Networks routers through a Telnet or secure shell connection.

- To open a Telnet session to a router, use the telnet operational mode command. For example:

  user@host> **telnet 10.10.10.3**

- To open a secure shell connection, use the ssh operational command. For example:

  user@host> **ssh host 10.10.10.3**

Configuring JUNOS Routing Platforms to Interact with the SAE

To configure the JUNOS routing platform to interact with the SAE:

1. Include the following statements at the [edit system services service-deployment] hierarchy level.

   ```
   [edit system services service-deployment]
   servers server-address {
   port port-number;
   }
   source-address source-address;
   ```

2. Use the following guidelines for the variables in these statements.

server-address

- Specifies the IP address of the host on which you install the SAE.
- Value—IP address
- Guidelines—Be sure this setting matches the corresponding value in the SAE configuration.
- Default—None
- Example—192.0.2.2

port-number

- Specifies the port number for the SAE.
- Value—TCP port number
- Guidelines—Be sure this setting matches the corresponding value in the SAE configuration.
- Default—3333
- Example—3333

source-address

- Specifies the IP address of the source that sends traffic to the SAE.
- Value—IP address
- Guidelines—This setting is optional.
- Default—None
- Example—192.0.2.2

Configuring the JUNOS Routing Platform to Apply Changes It Receives from the SAE

To configure the JUNOS routing platform to receive configuration statements from the SAE and apply those statements to the configuration:

1. Create a configuration group called sdx that contains the configuration statements that the SAE sends to the JUNOS routing platform. To do so, include the groups statement at the [edit] level, and specify the name sdx.

   ```
   [edit]
   groups {
       sdx;
   }
   ```

2. Configure the JUNOS routing platform to apply these statements to the configuration. To do so, include the apply-groups statement at the [edit] level.

   ```
   [edit]
   set apply-groups sdx;
   ```

Disabling Interactions Between the SAE and JUNOS Routing Platforms

To disable the SRC software process, enter the following command:

```
root@ui1#set system processes service-deployment disable
root@ui1#commit
```

When you disable the SRC software process, it is still available on the JUNOS routing platform.

To reenable the SRC software process, enter the following command:

```
root@ui1#delete system processes service-deployment disable
root@ui1#commit
```

The SRC software process attempts to reconnect the JUNOS routing platform to the SAE.

Monitoring Interactions Between the SAE and JUNOS Routing Platforms

Use the following command on JUNOS routing platforms to monitor the connection between the JUNOS routing platform and the SAE.

```
root@ui1> show system services service-deployment
Connected to 172.17.20.151 port 3333 since 2004-02-06 14:50:31 PST
Keepalive settings: Interval 15 seconds
Keepalives sent: 100, Last sent: 6 seconds ago
Notifications sent: 0
Last update from peer: 00:00:06 ago
```

You can also monitor the interactions between the SRC software and JUNOS routing platforms in the log files for the SAE and in the log files generated by the SRC software process on the JUNOS routing platform.

- For information about configuring logging for the SAE, see *SRC-PE Monitoring and Troubleshooting Guide, Chapter 3, Configuring Logging for SRC Components with the CLI*.
- For information about configuring logging on JUNOS routing platforms, see *JUNOS System Basics Configuration Guide*.

Troubleshooting Problems with the SRC Software Process with the C-Web Interface

To troubleshoot SRC problems on the JUNOS routing platform, review the log files for the SAE and the log files generated by the SRC software process on the router. If the log files indicate that the SRC software process on the JUNOS routing platform is not responding:

1. Look at the status of the process on the JUNOS routing platform.

   ```
   root@ui1>show system services service-deployment
   Connected to 172.17.20.151 port 3333 since 2004-02-06 14:50:31 PST
   Keepalive settings: Interval 15 seconds
   Keepalives sent: 100, Last sent: 6 seconds ago
   Notifications sent: 0
   Last update from peer: 00:00:06 ago
   ```

2. If you see the message “error: the service-deployment subsystem is not running,” reenable the SRC software process. See *Disabling Interactions Between the SAE and JUNOS Routing Platforms* on page 181.

3. If the process is already enabled, review the configurations of the router and the SAE in the directory, and fix any problems.

4. Restart the SRC software process on the router.

   ```
   root@ui1>restart service-deployment
   ```

   The SAE synchronizes with the SRC software process and deletes unnecessary data from the router.

Deleting All SRC Data on JUNOS Routing Platforms

If deleting parts of the SRC data on a JUNOS routing platform fails to solve problems, delete all the SRC data and restart the SRC software process. To do so:

1. Delete all SRC interfaces and services.

   delete groups sdx
   root@ui1#**commit**

2. If you are running SDX software releases 5.0 through 6.1, you should also delete interface sessions. (After release 6.2, session data is no longer stored on the router, it is stored on the SAE host using the session store feature.)

   delete groups sdx-sessions
   root@ui1#**commit**

3. Restart the SRC software process on the router.

   root@ui1>**restart service-deployment**

Viewing the State of JUNOS Device Drivers with the C-Web Interface

If the log files indicate a problem with a specific driver, review the configuration of the associated with the JUNOS device driver with the C-Web interface.

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

   The Drivers pane appears.

2. Enter information as described in the Help text in the main pane, and click **OK**.

   The Drivers pane displays information about the JUNOS device driver.

Viewing Statistics for Specific JUNOS Device Drivers with the C-Web Interface

To view SNMP statistics about a specific JUNOS device driver:

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

   The Device 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 JUNOS router drivers and PCMM drivers, use the format:

   default@<router name>

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

   The Device pane displays statistics for a specific JUNOS device driver.

Viewing Statistics for All JUNOS Device Drivers with the C-Web Interface

To view SNMP statistics for all JUNOS device drivers:

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

   The Common pane appears.

2. Enter information as described in the Help text in the main pane, and click **OK**.

   The Common pane displays statistics for the JUNOS device drivers.