Chapter 7

Configuring Remote Access to a C-series Controller with the SRC CLI

This chapter describes how to configure access to a C-series Controller.

You can also use the C-Web interface to configure access to a C-series Controller, See *SRC-PE C-Web Interface Configuration Guide, Chapter 1, Configuring Remote Access to a C-series Controller with the C-Web Interface*.

Topics in this chapter include:

Configuring External Interfaces on a C-series Controller

The C-series Controller provides the following interfaces:

- Serial port—9600 baud

  The serial port is enabled by default. You can use the serial port to connect to a console terminal and perform initial configuration as well as configuration updates.

- Two external Gigabit Ethernet interfaces—eth0 and eth1

  The eth0 interface is designed to provide access from a network that is behind a firewall. This interface accepts connections from protocols supported by the SRC software. When you configure an SRC component, the specified port is opened on this interface.

  The eth1 interface is designed to provide access for applications on an external network, such as the Internet. You can configure a limited number of ports on this interface. By default, no inbound ports are open.

- Optional two additional Gigabit Ethernet interfaces—eth2 and eth3

  These interfaces require an additional input/output module. You can obtain a module to support either RJ-45 or optical connections.

- Two USB interfaces

Configuring Gigabit Ethernet Interfaces for IPv4 with the SRC CLI

You can configure the Gigabit Ethernet interfaces to use IPv4 or IPv6 to allow remote access to the C-series Controller. You can specify an IP address with mask or a broadcast address with mask for an interface.

Use the following configuration statements to configure Gigabit Ethernet interfaces to use IP v4 and the [edit] hierarchy level:

```
interfaces name unit unit-number

interfaces name unit unit-number family inet {
    address address;
    broadcast broadcast;
}
```

To configure a Gigabit Ethernet interface to use IPv4:

1. From configuration mode, access the configuration statement that configures the interface.

   [edit]
   user@host# **edit interfaces name** *unit unit-number*

   where *unit-number* is a number that you can assign for a logical interface identifier.

   For example:

   [edit]
   user@host# **edit interfaces eth0**

2. Specify the unit, family, and IP address for the interface.

   [edit interfaces eth0]
   user@host# **set unit** *number* **family inet address** *address*

   For example, to configure an interface with only an IP address:

   [edit interfaces eth0]
   user@host# **set unit 0 family inet6 address 192.2.0.10/24**

3. Verify the interface configuration.

```
[edit interfaces eth0]
user@host# show
unit 0 {
  family {
    inet6 {
      address 192.2.0.10/24;
    }
  }
}
```

Configuring Gigabit Ethernet Interfaces for IPv6 with the SRC CLI

You can configure the Gigabit Ethernet interfaces to use IPv4 or IPv6 to allow remote access to the C-series Controller. You can specify an IP address with mask or a broadcast address with mask for an interface.

Use the following configuration statement to configure Gigabit Ethernet interfaces to use IPv6 at the [edit] hierarchy level:

interfaces name unit *unit-number* family inet6 address *address*;

To configure a Gigabit Ethernet interface to use IPv6:

1. From configuration mode, access the configuration statement that configures the interface.

   [edit]
   user@host# **edit interfaces name** *unit unit-number*

   where *unit-number* is a number that you can assign for a logical interface identifier.

   For example:

   [edit]
   user@host# **edit interfaces eth0**

2. Specify the unit, family, and IP address for the interface.

   [edit interfaces eth0]
   user@host# **set unit** *number* **family inet6 address** *address*

   For example:

   [edit interfaces eth0]
   user@host# **set unit 0 family inet6 address 2001:DB8:10AB:CD30::1/64**

3. Verify the interface configuration.

```
[edit interfaces eth0]
user@host# show
unit 0 {
  family {
    inet6 {
      address 10AB:0:0:CD30::/20;
    }
  }
}
```

Configuring Tunnel Interfaces with the SRC CLI

A tunnel allows direct connection between a remote location and an application running on the C-series Controller; a tunnel lets you use the redirect server in deployments where a JUNOSe router does not have a direct connection to the C-series Controller.

The C-series Controller supports the following types of tunnel interfaces:

- GRE—Generic routing encapsulation. Encapsulates traffic that can use various network protocols within IP. For C-series Controllers, the tunnel interface encapsulates IP packets.
- IP-over-IP—Encapsulates IP packets within IP packets.
- SIT—Encapsulates IPv6 traffic in an IPv4 tunnel. This type of tunnel allows compatibility of IPv6 traffic within an IPv4 network.

The other endpoint for the tunnel on a device must be configured for the tunnel to be operational.

The local address of a tunnel connection is an IP address that is configured for a unit (logical interface). Before you configure a tunnel interface, configure the interface on the C-series Controller.

See *Configuring Gigabit Ethernet Interfaces for IPv4 with the SRC CLI* on page 72.

Use the following configuration statements to configure tunnel interfaces at the [edit] hierarchy level:

```
interfaces name tunnel {
    mode (ipip | gre | sit);
    destination destination;
    source source;
    key key;
    interface interface;
    ttl ttl;
}

interfaces name unit unit-number family inet {
    address address;
}
```

To configure a tunnel interface on a C-series Controller:

1. From configuration mode, access the configuration statement that configures tunnel interfaces.

   ```
   [edit]
   user@host# edit interfaces name tunnel
   ```

   For example:

   ```
   [edit]
   user@host# edit interfaces ip-tunnel tunnel
   ```

2. Configure the type of tunnel.

   ```
   [edit interfaces ip-tunnel tunnel]
   user@host# set mode ipip
   ```

   or

   ```
   [edit interfaces ip-tunnel tunnel]
   user@host# set mode gre
   ```

   or

   ```
   [edit interfaces ip-tunnel tunnel]
   user@host# set mode sit
   ```

3. Specify the IP address of the remote end of the tunnel.

   [edit interfaces ip-tunnel tunnel]
   user@host# **set destination** *destination*

   For example:

   [edit interfaces ip-tunnel tunnel]
   user@host# **set destination 192.0.2.20**

4. (Optional) Specify an IP address that will not change for the local tunnel endpoint. It must be an address on another interface of this host.

   [edit interfaces ip-tunnel tunnel]
   user@host# **set source** *source*

   For example:

   [edit interfaces ip-tunnel tunnel]
   user@host# **set source 192.20.10.5**

5. (Optional) For a GRE tunnel, specify a key.

   [edit interfaces ip-tunnel tunnel]
   user@host# **set key** *key*

   For example:

   [edit interfaces ip-tunnel tunnel]
   user@host# **set key 250**

6. (Optional) Specify an existing physical interface on the C-series Controller.

   [edit interfaces ip-tunnel tunnel]
   user@host# **set interface** *interface*

   For example:

   [edit interfaces ip-tunnel tunnel]
   user@host# **set interface eth0**

7. (Optional) Specify the lifetime of tunneled packets.

   [edit interfaces ip-tunnel tunnel]
   user@host# **set ttl** *ttl*

   For example:

   [edit interfaces ip-tunnel tunnel]
   user@host# **set ttl 110**

8. Verify the configuration by running the show command. For example:

```
[edit interfaces]
user@host# show
unit 0 {
  family {
    inet6 {
      address 192.2.0.10/24;
    }
  }
}
ip-tunnel {
  tunnel {
      mode ipip;
      destination 192.0.2.20;
      source 192.20.10.5;
      interface eth0;
      ttl 110;
    }
}
```

Configuring a Static Route to Devices on Other Networks with the SRC CLI

In some instances, the SRC software might need to connect to devices that reside on networks other than the one that the SRC software accesses directly. You can configure a static route for the software to be able to connect devices on other networks.

When you specify IP addresses for a static route, include a network mask.

To configure a static route to another network:

- From configuration mode, enter the following command at the top level of the hierarchy.

  [edit]
  user@host# **set routing-options static route** *destination* **next-hop** *next-hop*

  The next-hop option is required.

You can also specify that packets to the specified destination be dropped and that an ICMP unreachable message be returned.

To specify that packets to a specified network be dropped:

- From configuration mode, enter the following command at the top level of the hierarchy.

  [edit]
  user@host# **set routing-options static route** *destination* **next-hop** *next-hop* **reject**

Securing Connections Between a C-series Controller and Remote Hosts

For security reasons, take care to limit the number of open ports you configure for applications and SRC components on the external interfaces. To review the default port settings for SRC components, see *Chapter 34, Defining an Initial Configuration on a Solaris Platform* which provides information about an initial configuration on a Solaris platform.

By default, SSH for nonroot users is enabled on C-series Controllers. Otherwise, you configure the C-series Controller to explicitly allow users on remote systems to access it. Table 9 lists the applications through which remote users can access a C-series Controller.

Table 9: Applications to Remotely Access the C-series Controller

Application	Information About Access Configuration
SSH	*Configuring a C-series Controller to Accept SSH Connections with the SRC CLI* on page 78
Telnet	*Configuring a C-series Controller to Accept Telnet Connections with the SRC CLI* on page 79
NETCONF	*Configuring a C-series Controller to Accept NETCONF Connections with the SRC CLI* on page 80
C-Web interface	*Chapter 6, Accessing and Using the C-Web Interface*
Policies, Services, and Subscribers CLI	*Chapter 5, Accessing and Starting the SRC CLI*

You can also configure security certificates for use by HTTPS connections.

You can connect from a C-series Controller to remote hosts through:

- SSH
- Telnet
- FTP by means of a file URL

Configuring a C-series Controller to Accept SSH Connections with the SRC CLI

You can enable SSH to let users who have the appropriate privileges connect to a C-series Controller. For security reasons, we recommend that you do not allow remote users to access the CLI as `root`.

Use the following configuration statements to enable SSH access from the [edit] hierarchy level:

```
system services ssh {
    root-login (allow | deny | deny-password);
    protocol-version (v1 | v2);
}
```

To configure the C-series Controller to accept SSH connections:

1. From configuration mode, access the [edit system services ssh] hierarchy level.
2. (Optional) Specify that SSH version 1 be used.

   ```
   [edit system services ssh]
   user@host> set protocol-version v1
   ```

   SSH version 2 is enabled by default.
3. (Optional) Specify whether or not to allow root login through SSH:

   ```
   [edit system services ssh]
   user@host> set root-login (allow | deny | deny-password)
   ```

   where:

   - allow—Allow users to log in to the C-series Controller as `root` through SSH.
   - deny—Disable users from logging in to the C-series Controller as `root` through SSH.
   - deny-password—Allow users to log in to the C-series Controller as `root` through SSH when the authentication method (for example, RSA authentication) does not require a password. (Default)

Configuring a C-series Controller to Accept Telnet Connections with the SRC CLI

You can enable Telnet to let users who have the appropriate privileges connect to a C-series Controller. The system does not allow `root` access over a Telnet connection.

Use the following configuration statements to enable Telnet access from the [edit] hierarchy level:

```
system services {
    telnet;
}
```

To configure the C-series Controller to accept Telnet connections:

```
[edit]
user@host# set system services telnet
```

Configuring a C-series Controller to Accept NETCONF Connections with the SRC CLI

Use the following configuration statements to enable NETCONF access from the [edit] hierarchy level:

```
system services netconf {
    ssh;
}
```

To configure the C-series Controller to accept NETCONF connections:

1. From configuration mode, access the [edit system services netconf] hierarchy level.

   ```
   [edit]
   user@host# edit system services netconf
   ```

2. (Optional) Enable NETCONF to run over SSH.

   ```
   [edit system services netconf]
   user@host# set ssh
   ```