Preparing Devices for Management

Before you start to manage a device using Security Manager, you should prepare the device with at least a minimal configuration. The following sections describe the basic device configurations needed for various transport protocols or device types. Before configuring transport protocols, determine the requirements for your devices by reading Understanding Device Communication Requirements, page 2-1.

Understanding Device Communication Requirements

Security Manager provides many different ways for you to manage devices. The easiest methods involve Security Manager directly contacting the devices. Security Manager might access a device during inventory or policy discovery, during configuration deployment, or in response to actions you take in Security Manager that request device contact (such as testing connectivity).

Because you can use off-line methods to add devices to the Security Manager inventory or to deploy configuration changes to the devices, configuring device communication settings for Security Manager's use is optional. However, you typically need to configure basic device communication settings on the devices to implement your off-line or customized configuration deployment tools.

In Security Manager, you can configure which transport protocol to use as the default for a type of device, and change it for specific devices that are configured to respond to a different protocol. Security Manager is configured with default protocols that are the most commonly-used protocols for that type of device. To change the default device communication setting for a type of device, select **Tools > Security Manager Administration** and select **Device Communication** from the table of contents (for more information, see Device Communication Page, page 11-21). To change the transport setting for a specific device, modify its device properties as described in Viewing or Changing Device Properties, page 3-40.

Security Manager can use these transport protocols:

- SSL (HTTPS)—Secure Socket Layer, which is an HTTPS connection, is the only transport protocol used with PIX Firewalls, Adaptive Security Appliances (ASA), and Firewall Services Modules (FWSM). It is also the default protocol for IPS devices and for routers running Cisco IOS Software release 12.3 or higher.

  If you use SSL as the transport protocol on Cisco IOS routers, you must also configure SSH on the routers. Security Manager uses SSH connections to handle interactive command deployments during SSL deployments.

  For information on configuring SSL, see Setting Up SSL (HTTPS), page 2-3.

- SSH—Secure Shell is the default transport protocol for Catalyst switches and Catalyst 6500/7600 devices. You can also use it with Cisco IOS routers.

  For information on configuring SSH, see Setting Up SSH, page 2-5.

- Telnet—Telnet is the default protocol for routers running Cisco IOS software releases 12.1 and 12.2. You can also use it with Catalyst switches, Catalyst 6500/7600 devices, and routers running Cisco IOS Software release 12.3 and higher. See the Cisco IOS software documentation for configuring Telnet.

- HTTP—You can use HTTP instead of HTTPS (SSL) with IPS devices. HTTP is not the default protocol for any device type.

- TMS—Token Management Server is treated like a transport protocol in Security Manager, but it is not a real transport protocol. Instead, by configuring TMS as the transport protocol of a router, you are telling Security Manager to deploy configurations to a TMS. From the TMS, you can download the configuration to an eToken, plug the eToken into the router's USB bus, and update the configuration. TMS is available only for certain routers running Cisco IOS Software 12.3 or higher.

  For information on deploying configurations to a TMS and downloading them to a router, see Deploying Configurations to a Token Management Server, page 8-42.

Security Manager can also use indirect methods to deploy configurations to devices, staging the configuration on a server that manages the deployment to the devices. These indirect methods also allow you to use dynamic IP addresses on your devices. The methods are not treated as transport protocols, but as adjuncts to the transport protocol for the device. You can use these indirect methods:

- AUS (Auto Update Server)—When you add a device to Security Manager, you can select the AUS server that is managing it. You can use AUS with PIX Firewalls and ASA devices.

  For information on configuring a device to use an AUS server, see Setting Up AUS or Configuration Engine, page 2-7.

- Configuration Engine—When you add a router to Security Manager, you can select the Configuration Engine that is managing it.

  For more information on configuring a router to use a Configuration Engine server, see Setting Up AUS or Configuration Engine, page 2-7.

For information on adding devices that use AUS or Configuration Engine servers to Security Manager, and how to add the servers, see these topics:

- Adding Devices to the Device Inventory, page 3-6
- Adding, Editing, or Deleting Auto Update Servers or Configuration Engines, page 3-36

Setting Up SSL (HTTPS)

With many devices, you can use the Secure Socket Layer (SSL) protocol, also known as HTTPS, to communicate with the device. When you deploy configurations with this protocol, Security Manager encrypts the configuration file before sending it to the device.

The following topics describe how to set up SSL on the devices:

Setting Up SSL (HTTPS) on PIX Firewall, ASA and FWSM Devices

This procedure describes the tasks to complete before you use SSL as the transport protocol for device management on PIX Firewall, ASA and FWSM devices.

Step 1 Enter configuration mode.

```
hostname# config terminal
```

Respond to the prompts appropriately. Here are some tips:

- Enter **y** when the prompt asks if you want to preconfigure using interactive prompts.
- Enter the current enable password.
- Specify the time zone, year, month, day, and time.
- If the device:
  - Is new—Specify the network interface IP address and network mask that applies to the inside IP address of the device.
  - Exists—Verify that the interface IP address and mask are correct.
- If the device:
  - Is new—Specify the hostname and the domain name.
  - Exists—Verify that the hostname and domain name are correct.
- When prompted for the IP address of the host that runs the PIX Device Manager, specify the IP address of the Security Manager server.
- Enter **yes** when the prompt asks if you want to write the above changes to Flash.

Step 2 If you are configuring an ASA, specify the SSL/TLS protocol version the ASA uses when acting as a server. Beginning with version 4.8, Security Manager supports all SSL/TLS protocol versions, the latest certified version being TLS 1.2.

```
hostname(config)# ssl server-version any
```

Step 3 Enable the HTTP server.

```
hostname(config)# http server enable
```

Step 4 Specify the host or network authorized to initiate an HTTP connection to the device.

```
hostname(config)# http ip_address  [netmask] [if_name]
```

Where:

- *ip_address*—The IP address of the Security Manager server.
- *netmask*—The network mask for the IP address.
- *if_name*—The device interface name (default is **inside**) from which Security Manager initiates the HTTP connection.

Step 5 Save the current configuration in Flash memory.

```
hostname(config)# write memory
```

Setting Up SSL on Cisco IOS Routers

This procedure describes the tasks to complete before you use SSL as the transport protocol for device management on Cisco IOS routers.

Step 1 Enter configuration mode.

```
hostname# config terminal
```

Step 2 Configure the hostname and domain name if the device is new.

```
router(config)# hostname name
hostname(config)# ip domain-name your_domain
```

Step 3 Configure level 15 privilege. SSL requires that you must have level 15 privileges to log in to a Cisco IOS router.

```
hostname(config)# username username privilege 15 password 0 password
```

Step 4 Enable either local authorization or AAA authorization:

- Local authorization— If you are using AAA for authorization but would like to use local authorization, use the following commands to disable AAA authorization and AAA authentication at login, where *list-name* is a character string used to name the list of authorization methods, and to enable local authorization using the username you just configured:

  ```
  hostname(config)# no aaa authorization network list-name
  hostname(config)# no aaa authentication login list-name
  hostname(config)#ip http authentication local
  ```

  If you do not enter the **ip http authentication local** command, the default enable password is used for authentication.

- AAA authorization—Use the following commands to enable AAA authentication and authorization. The last two commands are necessary only if multiple AAA lists are defined; *list-name* is a character string used to name the list of authorization methods. These commands authenticate the user that is contacting the device using the HTTPS protocol.

  ```
  hostname(config)#ip http authentication aaa
  hostname(config)#ip http authentication aaa login-authentication list-name
  hostname(config)# ip http authentication aaa exec-authorization list-name
  ```

Step 5 Enable the HTTPS server.

```
hostname(config)# ip http secure-server
```

Step 6 Exit configuration mode and return to Exec mode.

```
hostname(config)# exit
```

Step 7 Verify that SSL is set up on the device. The Device should respond with an "enabled" status.

```
hostname# show ip http server secure status
```

Setting Up SSH

You can use the Secure Shell (SSH) protocol to communicate with Cisco IOS Routers, Catalyst switches, and Catalyst 6500/7600 devices. This protocol provides strong authentication and secure communications over insecure channels. Security Manager supports both SSH versions 1.5 and 2. Once connected to the device, Security Manager determines which version to use and communicates using that version.

The following topics describe how to set up SSH on the supported devices:

- Critical Line-Ending Conventions for SSH, page 2-5
- Testing Authentication, page 2-5
- Setting Up SSH on Cisco IOS Routers, Catalyst Switches, and Catalyst 6500/7600 devices, page 2-6
- Preventing Non-SSH Connections (Optional), page 2-7

Critical Line-Ending Conventions for SSH

The following line-ending conventions for SSH must be observed to avoid system failure:

- Do not end banner message lines with "#", "# ", ">", or "> ". If your system requires a pound sign or greater-than sign at the end of a banner message, ensure that it is followed by two spaces.
- Do not use banner message lines that contain only "Username: " or "Password: "
- Do not customize the device user EXEC mode prompt to not end with ">" or "#".

Testing Authentication

Before you set up SSH, you must test authentication without SSH to make sure the device can be authenticated. You can authenticate with a local username and password or with an authentication, authorization, and accounting (AAA) server running TACACS+ or RADIUS.

This procedure describes how to test authentication without SSH using a local or AAA server username and password.

Step 1 Enter configuration mode.

```
router# config terminal
```

Step 2 Specify that the local username and password should be used in the absence of AAA statements. On Cisco IOS routers, you can use the **login local** command on VTY lines instead of the **aaa new-model** command.

```
hostname(config)#aaa new-model
```

Step 3 (Optional) Configure a user account in the local database of the device.

```
hostname(config)# username name password 0 password
```

Step 4 Exit configuration mode and return to Exec mode.

```
hostname(config)# exit
```

Step 5 Save the configuration changes.

```
hostname(config)# write memory
```

Setting Up SSH on Cisco IOS Routers, Catalyst Switches, and Catalyst 6500/7600 devices

This procedure describes the tasks required to set up SSH on Cisco IOS routers, Catalyst switches, and Catalyst 6500/7600 devices.

Tip You must configure SSH on Cisco IOS routers because Security Manager uses SSH connections to handle interactive command deployments during SSL deployments.

Related Topics

- Critical Line-Ending Conventions for SSH, page 2-5
- Testing Authentication, page 2-5
- Preventing Non-SSH Connections (Optional), page 2-7

Step 1 Enter configuration mode.

```
router# config terminal
```

Step 2 Configure the hostname and domain name if the device is new.

```
router(config)# hostname name
hostname(config)# ip domain-name your_domain
```

Step 3 Generate the RSA key pair for the SSH session. When the device prompts you to enter the size of the modulus, we recommend that you enter 1024.

```
hostname(config)# crypto key generate rsa
```

Step 4 (Optional) Set the timeout interval in minutes and the number of retries.

```
hostname(config)# ip ssh timeout time
hostname(config)# ip ssh authentication-retries n
```

Step 5 Exit configuration mode and return to Exec mode.

```
hostname(config)# exit
```

Step 6 Save the configuration changes.

```
hostname# write memory
```

Preventing Non-SSH Connections (Optional)

After configuring SSH, you can configure the Cisco IOS routers, Catalyst switches, and Catalyst 6500/7600 devices to use SSH connections only.

Related Topics

- Critical Line-Ending Conventions for SSH, page 2-5
- Testing Authentication, page 2-5
- Setting Up SSH on Cisco IOS Routers, Catalyst Switches, and Catalyst 6500/7600 devices, page 2-6

Step 1 Enter configuration mode.

```
router# config terminal
```

Step 2 Set up the router for Telnet access, specifying the first and last line numbers that can be used (numbers range from 0 to 1180, and the last number must be greater than the first number).

```
hostname(config)# line vty first_line last_line
```

Step 3 Prevent non-SSH connections, such as Telnet.

```
hostname(config-line)# transport input ssh
```

Step 4 Exit configuration mode.

```
hostname(config-line)# end
```

Step 5 Save the configuration changes.

```
hostname# write memory
```

Setting Up AUS or Configuration Engine

With many devices, you can use an intermediate transport server to stage configuration updates to the device. These transport servers can also allow you to manage devices that use dynamically assigned IP address (using a DHCP server) instead of static IP addresses. When you deploy configurations using a transport server, Security Manager deploys the configuration to the server, and the device retrieves the configuration from the server. You can use Auto Update Server, running the AUS protocol, or Cisco Configuration Engine, running the CNS protocol.

The following topics describe how to set up AUS or CNS on the devices:

- Setting Up AUS on PIX Firewall and ASA Devices, page 2-8
- Setting Up CNS on Cisco IOS Routers in Event-Bus Mode, page 2-8
- Setting Up CNS on Cisco IOS Routers in Call-Home Mode, page 2-9

Setting Up AUS on PIX Firewall and ASA Devices

You can configure PIX firewalls and ASA devices to use the AUS protocol to contact an Auto Update Server or CNS Configuration Engine for configuration and image updates. When using Configuration Engine, the device uses the same AUS protocol used for Auto Update Server, so the configuration is the same. For an end-to-end explanation of how AUS/CE deployment works, see Deploying Configurations Using an Auto Update Server or CNS Configuration Engine, page 8-41.

You need to initially configure AUS settings on the device so that the device knows that it must contact the AUS/CE server for configuration updates. After the initial deployment, you can change these settings using the **Platform > Device Admin > Server Access > AUS** policy.

This procedure describes the tasks to complete before you use AUS or CNS as the transport protocol for device management on PIX firewall and ASA devices.

Step 1 Enter configuration mode.

```
router# config terminal
```

Step 2 Connect to the AUS. Specify a username and its password that can log into Security Manager. The port number is typically 443.

```
hostname(config)# auto-update server https:// username:password@AUSserver_IP_address:port
/autoupdate/AutoUpdateServlet
```

Step 3 Specify the polling period for AUS.

```
hostname(config)# auto-update poll-period  poll_period  [retry_count ] [retry_period ]
```

Where:

- *poll_period*—The polling period interval between two updates. Default is 720 minutes (12 hours).
- *retry_count*—(Optional) The number of times to retry if the server connection attempt fails. Default is 0.
- *retry_period*—(Optional) The number of minutes between retries. Default is 5.

Step 4 Configure the device to use the specified unique device ID to identify itself.

```
hostname(config)# auto-update device-id  [ hardware-serial | hostname |
ipaddress  [if_name ] |  mac-address [if_name ] | string text ]
```

Where:

- *if_name*—The device interface name (the default is **inside**).
- *text*—A unique string name.

Step 5 Save the configuration changes.

```
hostname# write memory
```

Setting Up CNS on Cisco IOS Routers in Event-Bus Mode

You can configure Cisco IOS routers to use the CNS protocol to contact a Cisco Configuration Engine for configuration and image updates. The Configuration Engine can operate in two modes, event-bus and call-home. The following procedure describes how to configure a router to use event-bus mode. For information on using call-home mode, see Setting Up CNS on Cisco IOS Routers in Call-Home Mode,

page 2-9.

See the Configuration Engine product documentation for more information about configuring and using the product.

Step 1 Enter configuration mode.

```
router# config terminal
```

Step 2 Configure the hostname and domain name if the device is new.

```
router(config)# hostname name
hostname(config)# ip domain-name your_domain
```

Step 3 Specify the trusted server for the CNS agent. Enter the IP address of the trusted server.

```
hostname(config)# cns trusted-server all-agents ip_address
```

Step 4 Configure the CNS event gateway, which provides CNS event services to Cisco IOS clients. Enter the IP address of the event gateway, and optionally the port. The default port is either 11011 (with no encryption) or 11012 (with encryption). Include the **encrypt** keyword to use an SSL encrypted link to the event gateway.

```
hostname(config)# cns event ip_address  [encrypt] [port]
```

Step 5 Start the CNS configuration agent and accept a partial configuration. Include the **encrypt** keyword to use an SSL encrypted link to the web server.

```
hostname(config)# cns config partial ip_address [encrypt]
```

Step 6 Set the CNS password, which must be the same password configured on the CNS gateway. For information on how to authenticate a Cisco IOS router on a Configuration Engine, see *Cisco Configuration Engine Administrator Guide* at http://www.cisco.com/en/US/products/sw/netmgtsw/ps4617/prod_maintenance_guides_list.html.

```
hostname(config)# cns password password
```

Step 7 Enable and configure the CNS execute agent. Include the **encrypt** keyword to use an SSL encrypted link to the exec server. You can specify a port number for the encrypted exchange if you do not want to use the default port 443.

```
hostname(config)# cns exec [encrypt [port]]
```

Step 8 Exit configuration mode and return to Exec mode.

```
hostname(config)# exit
```

Step 9 Save the configuration changes.

```
hostname# write memory
```

Setting Up CNS on Cisco IOS Routers in Call-Home Mode

You can configure Cisco IOS routers to use the CNS protocol to contact a Cisco Configuration Engine for configuration and image updates. The Configuration Engine can operate in two modes, event-bus and call-home. The following table describes the tasks to complete to configure a router to use call-home mode. For information on using event-bus mode, see Setting Up CNS on Cisco IOS Routers in Event-Bus Mode, page 2-8.

See the Configuration Engine product documentation for more information about configuring and using the product.

Step 1 Enter configuration mode.

```
router# config terminal
```

Step 2 Configure the hostname and domain name if the device is new.

```
router(config)# hostname name
hostname(config)# ip domain-name your_domain
```

Step 3 Specify the IP address of the trusted server for the CNS agent.

```
hostname(config)# cns trusted-server all-agents ip_address
```

Step 4 Specify schedule parameters for a Command Scheduler occurrence and enter kron-occurrence configuration mode.

```
hostname(config)# kron occurrence occurrence-name [user username ]
{in [[numdays:]numhours:]nummin  | at hours:min [[month] day-of-month] [day-of-week]} {oneshot  | recurring}
```

Where:

- *occurrence-name*—The name of the occurrence. The name can be from 1 to 31 characters. If the occurrence-name is new, an occurrence structure is created. If the occurrence-name is not new, the existing occurrence is edited.
- *username*—(Optional) The name of the user.
- **in** [*[numdays:]numhours:]nummin*—The occurrence should run after waiting the specified time. You can enter a number of days, hours, or minutes, or a combination of them. The timer starts when the occurrence is configured.
- **at** *hours:min [[month] day-of-month] [day-of-week]*—The occurrence should run at the specified hour and minute on the specified month and day, or day of the week. Specify the hour using the 24-hour clock.
- **oneshot**—Specifies that the occurrence is to run only once. After the occurrence runs, the configuration is removed.
- **recurring**—Specifies that the occurrence is to run on a recurring basis.

Step 5 Specify the policy list associated with a Command Scheduler occurrence. The name can be 1 to 31 characters. If the list-name is new, a policy list structure is created. If the list-name is not new, the existing policy list is edited.

Use the kron occurrence and policy-list commands to schedule one or more policy lists to run at the same time or interval.

```
hostname(config-kron-occurrence)# policy-list list-name
```

Step 6 Exit kron-occurrence and return to configuration mode.

```
hostname(config-kron-occurrence)# exit
```

Step 7 Specify a name for a Command Scheduler policy and enter kron-policy configuration mode. The name can be 1 to 31 characters. If the list-name is new, a policy list structure is created. If the list-name is not new, the existing policy list is edited.

```
hostname(config)# kron policy-list list-name
```

Step 8 Retrieve the configuration from the staged CNS job. Specify the IP address of the CNS server. You must use **JobbedDynaConfig** status so that the device retrieves the config from the staged CNS job; otherwise, the device retrieves the template associated with the device.

```
hostname(config-kron-policy)# cli cns config retrieve ip_address
 page /cns/JobbedDynaConfig status http:// ip_address /cns/PostStatus
```

Step 9 Exit kron-policy configuration mode and return to configuration mode.

```
hostname(config-kron-policy)# exit
```

Step 10 Enable and configure the CNS execute agent.

```
hostname(config)# cns exec
```

Step 11 Exit configuration mode and return to Exec mode.

```
hostname(config)# exit
```

Step 12 Save the configuration changes.

```
hostname# write memory
```

Configuring Licenses on Cisco ASA Devices

Devices that run Cisco ASA Software require Product Activation Keys for each feature license. Some licenses are optional, such as Botnet Traffic Filtering, and can be time-based. Other features are standard on some models, but optional on others, such as the Failover license, which is optional on the 5505 and 5510 models but standard on all other models.

You cannot install or activate ASA licenses through Security Manager. Instead, use the Adaptive Security Device Manager (ASDM). Enter the activation keys by selecting **Configuration > Device Management > Licensing > Activation Key** and following the instructions in the online help for that page. The Activation Key page also lists the state of all feature licenses. The ASDM online help includes extensive information about ASA licensing.

When you deploy configurations from Security Manager, the device must have active licenses for all features in the configuration or you will see deployment errors. In most cases, Security Manager does not prevent you from configuring a feature based on the licenses that are active on a device. For example, you can configure Botnet Traffic Filtering for a device even if that device has a disabled Botnet license.

The exception is the Failover license on the 5505 and 5510 models. There is a device property that you can set to indicate whether there is an active Failover license on a device: License Supports Failover. You can set this property by double-clicking the device (in Device view) to open the Device Properties page; the option is on the General tab (see Device Properties: General Page, page 3-41). If you discover policies on the device, for example, when adding the device to the inventory using the Add Device From Network or Add Device from File (from an inventory file, not a configuration file) options, Security Manager determines the state of the Failover license and sets the property appropriately. You are responsible for ensuring that the property remains accurate. You will see deployment failures if the property is selected but the device has an inactive Failover license.

Tip If you add the device using the New Device or Configuration File options, you can set the License Supports Failover property while adding the device instead of waiting to set it in the device properties.

Configuring Licenses on Cisco IOS Devices

Devices that run Cisco IOS Software require license files for various features, including security features. If these licenses are not installed on the device (such as the securityk9 package), Security Manager cannot configure commands that require a particular license level, and you will experience deployment failures when you try to deploy your policies to an unlicensed device.

Although you can use Security Manager to deploy and manage IPS licenses, you cannot use it to deploy and manage any other type of license. Configure these licenses directly on the device using the command line interface or use Cisco License Manager. Following is the general process for configuring licenses. For more information about configuring licenses, see Cisco IOS Software Activation Command Guide and Cisco IOS Software Activation Command Reference on Cisco.com.

1. Obtain the licenses required for the features you want to use or you can use the evaluation licenses that come bundled with some devices. Use the **show license all** command to view the available licenses.
2. Copy the purchased licenses to the flash storage on the device or put them on a TFTP server. For example, you could place the licenses on a TFTP server and use the **copy tftp flash0:** command to copy the files to the flash0 storage area.
3. Use the **license install** command to install each purchased license. For example:

   license install flash0:uc-base-CISCO2951-FHH1216P06Z.xml

   Some licenses prompt you to read and accept a license agreement.

   If you want to use an evaluation license, use the **license boot** command to enable them and then reload the device. You must accept the end-user license agreement before Security Manager can deploy configurations to the device.
4. You can use the **show version**, **show license feature**, and **show license all** commands to check on your installed licenses.

Initializing IPS Devices

To initialize an IPS device, you must configure the following settings. These are network settings, and only a user with administrator privileges on the IPS device can configure them:

- Sensor name
- IP address
- Netmask
- Default route
- Enable TLS/SSL (to enable TLS/SSL in the web server on the device)
- Web server port
- Use default ports

You configure these settings through the **setup** command in Intrusion Prevention System Device Manager (IDM) or in a command-line session, depending upon which platform is used by your IPS device. For a list of supported IPS platforms, see the supported devices and software versions information at the following URL: http://www.cisco.com/en/US/products/ps6498/products_device_support_tables_list.html.

For detailed information on these settings, refer to the technical documentation for your IPS device.

Note For information on preparing an IOS IPS device for use, see Initial Preparation of a Cisco IOS IPS Router, page 45-5.