Cisco Networking Services Configuration Guide, Cisco IOS Release 15S

First Published: November 05, 2012

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
     800 553-NETS (6387)
Fax: 408 527-0883
THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: http://www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2014 Cisco Systems, Inc. All rights reserved.
<table>
<thead>
<tr>
<th>CHAPTER 5</th>
<th>Cisco Networking Services Config Retrieve Enhancement with Retry and Interval</th>
<th>43</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Finding Feature Information</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>Information About CNS Config Retrieve Enhancement with Retry and Interval</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>Cisco Networking Services Config Retrieve Enhancement with Retry and Interval</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>How to Configure CNS Config Retrieve Enhancement with Retry and Interval</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Retrieving a Cisco Networking Services Configuration from a Server</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Configuration Examples for CNS Config Retrieve Enhancement with Retry and Interval</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>Example: Retrieving a Cisco Networking Services Configuration from a Server</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>Additional References</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>Feature Information for CNS Config Retrieve Enhancement with Retry and Interval</td>
<td>47</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER 6</th>
<th>Cisco Networking Services Interactive CLI</th>
<th>49</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Finding Feature Information</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>Information About CNS Interactive CLI</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>Cisco Networking Services Interactive CLI</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>Additional References</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Feature Information for CNS Interactive CLI</td>
<td>50</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER 7</th>
<th>Cisco Networking Services Security Enhancement</th>
<th>53</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Finding Feature Information</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td>Information About Cisco Networking Services Security Enhancement</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>Cisco Networking Services Security Enhancement</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>Cisco Networking Services Trusted Servers</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>How to Configure Cisco Networking Services Security Enhancement</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>Configuring Cisco Networking Services Trusted Servers</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>Configuration Examples for Cisco Networking Services Security Enhancement</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>Example: Configuring Cisco Networking Services Trusted Servers</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>Additional References</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>Feature Information for Cisco Networking Services Security Enhancement</td>
<td>57</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER 8</th>
<th>Command Scheduler (Kron)</th>
<th>59</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Finding Feature Information</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>Restrictions for Command Scheduler</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>Information About Command Scheduler (Kron)</td>
<td>60</td>
</tr>
</tbody>
</table>
Command Scheduler 60
How to Configure Command Scheduler (Kron) 60
   Configuring Command Scheduler Policy Lists and Occurrences 60
   Troubleshooting Tips 63
Configuration Examples for Command Scheduler (Kron) 63
   Example: Command Scheduler Policy Lists and Occurrences 63
Additional References 64
Feature Information for Command Scheduler (Kron) 65

CHAPTER 9
Network Configuration Protocol 67
   Finding Feature Information 67
   Prerequisites for NETCONF 67
   Information About NETCONF 68
      NETCONF Notifications 68
   How to Configure NETCONF 68
      Configuring the NETCONF Network Manager Application 68
      Delivering NETCONF Payloads 69
      Formatting NETCONF Notifications 71
      Monitoring and Maintaining NETCONF Sessions 74
   Configuration Examples for NETCONF 75
      Example: Configuring the NETCONF Network Manager Application 75
      Example: Monitoring NETCONF Sessions 76
   Additional References for NETCONF 79
   Feature Information for NETCONF 80
   Glossary 80

CHAPTER 10
NETCONF over SSHv2 83
   Finding Feature Information 83
   Prerequisites for NETCONF over SSHv2 83
   Restrictions for NETCONF over SSH 84
   Information About NETCONF over SSHv2 84
      NETCONF over SSHv2 84
   How to Configure NETCONF over SSHv2 85
      Enabling SSH Version 2 Using a Hostname and Domain Name 85
      Enabling SSH Version 2 Using RSA Key Pairs 86
<table>
<thead>
<tr>
<th>Chapter 11</th>
<th>NETCONF Access for Configurations over BEEP</th>
<th>97</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Finding Feature Information</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>Prerequisites for NETCONF Access for Configurations over BEEP</td>
<td>98</td>
</tr>
<tr>
<td></td>
<td>Restrictions for NETCONF Access for Configurations over BEEP</td>
<td>98</td>
</tr>
<tr>
<td></td>
<td>Information About NETCONF Access for Configurations over BEEP</td>
<td>98</td>
</tr>
<tr>
<td></td>
<td>NETCONF over BEEP Overview</td>
<td>98</td>
</tr>
<tr>
<td></td>
<td>How to Configure NETCONF Access for Configurations over BEEP</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td>Configuring an SASL Profile</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td>Enabling NETCONF over BEEP</td>
<td>101</td>
</tr>
<tr>
<td></td>
<td>Configuration Examples for NETCONF Access for Configurations over BEEP</td>
<td>104</td>
</tr>
<tr>
<td></td>
<td>Example: Enabling NETCONF over BEEP</td>
<td>104</td>
</tr>
<tr>
<td></td>
<td>Additional References for NETCONF Access for Configurations over BEEP</td>
<td>104</td>
</tr>
<tr>
<td></td>
<td>Feature Information for NETCONF Access for Configurations over BEEP</td>
<td>105</td>
</tr>
</tbody>
</table>
Configuring Cisco Networking Services

The Cisco Networking Services (CNS) feature is a collection of services that can provide remote event-driven configuring of Cisco IOS networking devices and remote execution of some command-line interface (CLI) commands.

- Finding Feature Information, page 1
- Prerequisites for Cisco Networking Services, page 1
- Restrictions for Cisco Networking Services, page 2
- Information About Cisco Networking Services, page 3
- How to Configure Cisco Networking Services, page 8
- Configuration Examples for Cisco Networking Services, page 16
- Additional References, page 19
- Feature Information for Cisco Networking Services, page 20

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Cisco Networking Services

- Configure the remote device to support the Cisco Networking Services configuration agent and the Cisco Networking Services event agent.
• Configure a transport protocol on the remote device that is compatible with the remote device’s external interface. The following table lists the supported transport protocols that can be used depending on the device interface.

• Create the configuration template in the Cisco Networking Services configuration-engine provisioning database. (This task is best done by a senior network designer.)

Table 1: Device Interface and Transport Protocols Required by Cisco Networking Services Services

<table>
<thead>
<tr>
<th>Device Interface</th>
<th>SLARP Transport Protocol</th>
<th>ATM InARP Transport Protocol</th>
<th>PPP (IPCP) Transport Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>ADSL</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Serial</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Restrictions for Cisco Networking Services

Cisco Networking Services Configuration Engine

• The Cisco Networking Services configuration engine must be the Cisco Intelligence Engine 2100 (Cisco IE2100) series and must be running software version 1.3.

• The configuration engine must have access to an information database of attributes for building a configuration. This database can reside on the Cisco IE2100 itself.

• Configuration templates must be prepared on the Cisco Networking Services configuration engine before installation of the remote device.

• The user of Cisco Networking Services Flow-Through Provisioning and the Cisco Networking Services configuration engine must be familiar with designing network topologies, designing configuration templates, and using the Cisco Networking Services configuration engine.

Remote Device

• The remote device must run a Cisco IOS image that supports the Cisco Networking Services configuration agent and Cisco Networking Services event agent.

• Ports must be prepared on the remote device for connection to the network.

• You must ensure that the remote device is configured using Cisco Configuration Express.
Information About Cisco Networking Services

Cisco Networking Services

Cisco Networking Services is a foundation technology for linking users to networking services and provides the infrastructure for the automated configuration of large numbers of network devices. Many IP networks are complex with many devices, and each device must currently be configured individually. When standard configurations do not exist or have been modified, the time involved in initial installation and subsequent upgrading is considerable. The volume of smaller, more standardized, customer networks is also growing faster than the number of available network engineers. Internet service providers (ISPs) now need a method for sending out partial configurations to introduce new services. To address all these issues, Cisco Networking Services has been designed to provide "plug-and-play" network services using a central directory service and distributed agents. Cisco Networking Services features include Cisco Networking Services configuration and event agents and a Flow-Through Provisioning structure. The configuration and event agents use a Cisco Networking Services configuration engine to provide methods for automating initial Cisco device configurations, incremental configurations, and synchronized configuration updates, and the configuration engine reports the status of the configuration load as an event to which a network monitoring or workflow application can subscribe. The Cisco Networking Services Flow-Through Provisioning uses the Cisco Networking Services configuration and event agents to provide an automated workflow, eliminating the need for an on-site technician.

Cisco Networking Services EXEC Agent

The CNS EXEC agent allows a remote application to execute an EXEC mode CLI command on a Cisco device by sending an event message that contains the command. A restricted set of EXEC show commands is supported.

Cisco Networking Services Results Messages

When a partial configuration has been received by the device, each line of the configuration will be applied in the same order as it was received. If the Cisco parser has an error with one of the lines of the configuration, then all the configuration up to this point will be applied to the device, but none of the configuration beyond the error will be applied. If an error occurs, the cns config partial command will retry until the configuration successfully completes. In the pull mode, the command will not retry after an error. By default, NVRAM will be updated except when the no-persist keyword is configured.

A message will be published on the Cisco Networking Services event bus after the partial configuration is complete. The Cisco Networking Services event bus will display one of the following status messages:

- cisco.mgmt.cns.config.complete—Cisco Networking Services configuration agent successfully applied the partial configuration.
- cisco.mgmt.cns.config.warning—Cisco Networking Services configuration agent fully applied the partial configuration, but encountered possible semantic errors.
- cisco.mgmt.cns.config.failure (CLI syntax)—Cisco Networking Services configuration agent encountered a command line interface (CLI) syntax error and was not able to apply the partial configuration.
• cisco.mgmt.cns.config.failure (CLI semantic)—Cisco Networking Services configuration agent encountered a CLI semantic error and was not able to apply the partial configuration.

With the CNS Enhanced Results Messages feature, a second message is sent to the subject “cisco.cns.config.results” in addition to the appropriate message above. The second message contains both overall and line-by-line information about the configuration that was sent and the result of the action requested in the original message. If the action requested was to apply the configuration, then the information in the results message is semantic in nature. If the action requested was to check syntax only, then the information in the results message is syntactical in nature.

Cisco Networking Services Message Formats

SOAP Message Format

Using the Service-Oriented Access Protocol (SOAP) protocol provides a way to format the layout of Cisco Networking Services messages in a consistent manner. SOAP is a lightweight protocol intended for exchanging structured information in a decentralized, distributed environment. SOAP uses extensible markup language (XML) technologies to define an extensible messaging framework that provides a message format that can be exchanged over a variety of underlying protocols.

Within the SOAP message structure, there is a security header that enables Cisco Networking Services notification messages to authenticate user credentials.

Cisco Networking Services messages are classified into three message types: request, response and notification. The formats of these three message types are defined below.

Request Message

The following is the format of a Cisco Networking Services request message to the Cisco device:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://www.w3.org/2003/05/soap-envelope">
  <SOAP:Header>
      SOAP:mustUnderstand="0">
      <wsse:usernameToken>
        <wsse:Username>john</wsse:Username>
        <wsse:Password>cisco</wsse:Password>
      </wsse:usernameToken>
    </wsse:Security>
  </SOAP:Header>
    <config-event config-action="read" no-syntax-check="TRUE">
      <config-data>
        <config-id>AAA</config-id>
        <cli>access-list 1 permit any</cli>
      </config-data>
    </config-event>
  </SOAP:Body>
</SOAP:Envelope>
```
The ReplyTo field is optional. In the absence of the ReplyTo field, the response to the request will be sent to the destination where the request originated. The body portion of this message contains the payload and is processed by the Cisco Networking Services agent mentioned in the Agent field.

**Response Message**

The following is the format of a Cisco Networking Services response message from the Cisco device as a response to a request:

```xml
<?xml version="1.0" encoding="UTF-8"?
SOAP:Envelope xmlns:SOAP="http://www.w3.org/2003/05.soap-envelope"
SOAP:Header
    SOAP:mustUnderstand="true"
    wsse:UsernameToken
        wsse:Username infysj-7204-8 /wsse:Username
        wsse:Password NTM3NTg2NzIzOTg2MTk2MjgzNQ== /wsse:Password
    /wsse:UsernameToken /wsse:Security
CNS:Agent CNS_CONFIG /CNS:Agent
CNS:Response
CNS:correlationID IDENTIFIER /CNS:correlationID
    CNS:Response
    /CNS:cnsHeader
    /SOAP:Header
    config-success config-id AAA /config-id /config-success
    /SOAP:Body
    /SOAP:Envelope
```

The value of CorrelationId is echoed from the corresponding request message.

The body portion of this message contains the response from the Cisco device to a request. If the request is successfully processed, the body portion contains the value of the response put in by the agent that processed the request. If the request cannot be successfully processed, then the body portion will contain an error response.

**Notification Message**

The following is the format of a Cisco Networking Services notification message sent from the Cisco device:

```xml
<?xml version="1.0" encoding="UTF-8"?
SOAP:Envelope xmlns:SOAP="http://www.w3.org/2003/05.soap-envelope"
SOAP:Header
    SOAP:mustUnderstand="true"
    wsse:UsernameToken
        wsse:Username dvlpr-7200-2 /wsse:Username
        wsse:Password /wsse:Password
    /wsse:UsernameToken /wsse:Security
CNS:Agent CNS_CONFIG_CHANGE/CNS:Agent
CNS:Notify /CNS:Notify
    CNS:Time 2006-01-09T18:57:08.441Z /CNS:Time
    /CNS:cnsHeader
    /SOAP:Header
    configChanged version="1.1" sessionData="complete"
    sequence lastReset="2005-12-11T20:18:39.673Z" 7 /sequence
```
A notification message is sent from the Cisco device without a corresponding request message when a configuration change is made. The body of the message contains the payload of the notification and it may also contain error information. If the request message sent to the Cisco device fails in XML parsing and the CorrelationId field cannot be parsed, then an error notification message will be sent instead of an error response.

Error Reporting

Error is reported in the body of the response or a notification message in the SOAP Fault element. The following is the format for reporting errors.

```xml
<?xml version="1.0" encoding="UTF-8"?
SOAP:Envelope xmlns:SOAP="http://www.w3.org/2003/05/soap-envelope"
SOAP:Header
SOAP:mustUnderstand="true"
wsse:UsernameToken
wsse:Username dvlpr-7200-2 /wsse:Username
wsse:Password /wsse:Password
/wsse:UsernameToken
/wsse:Security
CNS:Agent CNS_CONFIG /CNS:Agent
CNS:Response
CNS:correlationID SOAP_IDENTIFIER /CNS:correlationID
/CNS:Response
CNS:Time 2006-01-09T19:10:10.009Z /CNS:Time
/CNS:cnsHeader
/SOAP:Header
SOAP:Detail
config-failure
cfg-id AAA /config-id
erroinfo
line-number 1 /line-number
error-message CNS_INVALID_CLI_CMD /error-message
erroinfo
/config-failure
/SOAP:Detail
/SOAP:Fault
/SOAP:Body
/SOAP:Envelope
```
Cisco Networking Services IDs

The Cisco Networking Services ID is a text string that is used exclusively with a particular Cisco Networking Services agent. The Cisco Networking Services ID is used by the Cisco Networking Services agent to identify itself to the server application with which it communicates. For example, the Cisco Networking Services configuration agent will include the configuration ID when communicating between the networking device and the configuration server. The configuration server uses the Cisco Networking Services configuration ID as a key to locate the attribute containing the Cisco CLI configuration intended for the device that originated the configuration pull.

The network administrator must ensure a match between the Cisco Networking Services agent ID as defined on the routing device and the Cisco Networking Services agent ID contained in the directory attribute that corresponds to the configuration intended for the routing device. Within the routing device, the default value of the Cisco Networking Services agent ID is always set to the hostname. If the hostname changes, the Cisco Networking Services agent ID also changes. If the Cisco Networking Services agent ID is set using the CLI, any change will be followed by a message sent to syslog or an event message will be sent.

The Cisco Networking Services agent ID does not address security issues.

Cisco Networking Services Password

The Cisco Networking Services password is used to authenticate the Cisco Networking Services device. You must configure the Cisco Networking Services password the first time a device is deployed, and the Cisco Networking Services password must be the same as the bootstrap password set on the Configuration Engine (CE). If both the device and the CE bootstrap password use their default settings, a newly deployed device will be able to connect to the CE. Once connected, the CE manages the Cisco Networking Services password. Network administrators must ensure not to change the Cisco Networking Services password. If the Cisco Networking Services password is changed, connectivity to the CE will be lost.

Cisco Networking Services Zero Touch

The Cisco Networking Services Zero Touch feature provides a zero touch deployment solution where the device contacts a Cisco Networking Services configuration engine to retrieve its full configuration automatically. This capability is made possible through a single generic bootstrap configuration file common across all service provider end customers subscribing to the services. Within the Cisco Networking Services framework, customers can create this generic bootstrap configuration without device-specific or network-specific information such as interface type, line type, or controller type (if applicable).

The Cisco Networking Services connect functionality is configured with a set of Cisco Networking Services connect templates. A Cisco Networking Services connect profile is created for connecting to the Cisco Networking Services configuration engine and to implement the Cisco Networking Services connect templates on a Customer Premise Equipment (CPE) device. Cisco Networking Services connect variables can be used as placeholders within a Cisco Networking Services connect template configuration. These variables, such as the active DLCI, are substituted with real values before the Cisco Networking Services connect templates are sent to the device's parser.

To use the zero touch functionality, the device that is to be initialized must have a generic bootstrap configuration. This configuration includes Cisco Networking Services connect templates, Cisco Networking Services connect profiles, and the `cns config initial` command. This command initiates the Cisco Networking Services connect function.
The Cisco Networking Services connect functionality performs multiple ping iterations through the device’s interfaces and lines, as well as any available controllers. For each iteration, the Cisco Networking Services connect function attempts to ping the Cisco Networking Services configuration engine. If the ping is successful, the pertinent configuration information can be downloaded from the Cisco Networking Services configuration engine. If connectivity to the Cisco Networking Services configuration engine is unsuccessful, the Cisco Networking Services connect function removes the configuration applied to the selected interface, and the Cisco Networking Services connect process restarts with the next available interface specified by the Cisco Networking Services connect profile.

The Cisco Networking Services Zero Touch feature provides the following benefits:

- Ensures consistent Cisco Networking Services commands.
- Use of a channel service unit (E1 or T1 controller) is allowed.

## How to Configure Cisco Networking Services

### Deploying the Cisco Networking Services Device

Incremental or partial configuration allows the remote device to be incrementally configured after its initial configuration. You must perform these configurations manually through the Cisco Networking Services configuration engine. The registrar allows you to change the configuration templates, edit parameters, and submit the new configuration to the device without a software or hardware restart.

**Before You Begin**

Perform this task to manually install an initial Cisco Networking Services configuration.

Your remote device arrives from the factory with a bootstrap configuration. Upon initial power-on, the device automatically pulls a full initial configuration from the Cisco Networking Services configuration engine, although you can optionally arrange for this manually as well. After initial configuration, you can optionally arrange for periodic incremental (partial) configurations for synchronization purposes.

For more details on using the Cisco CNS configuration engine to automatically install the initial CNS configuration, see the Cisco CNS Configuration Engine Administrator’s Guide at http://www.cisco.com/en/US/docs/net_mgmt/configuration_engine/1.3/administration/guide/ag13.html

### Initial Cisco Networking Services Configuration

Initial configuration of the remote device occurs automatically when the device is initialized on the network. Optionally, you can perform this configuration manually.

Cisco Networking Services assigns the remote device a unique IP address or hostname. After resolving the IP address (using Serial Line Address Resolution Protocol (SLARP), ATM Inverse ARP (ATM InARP), or PPP protocols), the system optionally uses Domain Name System (DNS) reverse lookup to assign a hostname to the device and invokes the Cisco Networking Services agent to download the initial configuration from the Cisco Networking Services configuration engine.

### Incremental Configuration

Before you can configure an incremental configuration, Cisco Networking Services must be operational and the required Cisco Networking Services agents configured.
SUMMARY STEPS

1. enable
2. configure terminal
3. cns template connect name
4. cli config-text
5. Repeat Step 4 to add all required CLI commands.
6. exit
7. cns connect name [retry-interval interval-seconds] [retries number-retries] [timeout timeout-seconds] [sleep sleep-seconds]
8. Do one of the following:
   • discover {line line-type | controller controller-type | interface [interface-type]}
   • template name
9. exit
10. cns config initial {host-name | ip-address} [encrypt] [port-number] [page page] [syntax-check] [no-persist] [source interface name] [status url] [event] [inventory]
11. exit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Step 2 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td>Step 3 cns template connect name</td>
<td>Enters Cisco Networking Services template-connect configuration mode and defines the name of a Cisco Networking Services connect template.</td>
</tr>
<tr>
<td>Example: Device(config)# cns template connect template 1</td>
<td></td>
</tr>
<tr>
<td>Step 4 cli config-text</td>
<td>Specifies commands to configure the interface.</td>
</tr>
<tr>
<td>Example: Device(config-templ-conn)# cli encapsulation ppp</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>Repeat Step 4 to add all required CLI commands. Repeat Step 4 to add other CLI commands to configure the interface or to configure the modem lines.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-templ-conn)# cli ip directed-broadcast</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>Exits Cisco Networking Services template connect configuration mode and completes the configuration of a Cisco Networking Services connect template.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-templ-conn)# exit</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>Entering the <code>exit</code> command is required. This requirement was implemented to prevent accidentally entering a command without the <code>cli</code> command.</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>Enters Cisco Networking Services connect configuration mode and defines the parameters of a Cisco Networking Services connect profile for connecting to the Cisco Networking Services configuration engine.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config)# cns connect profile-1 retry-interval 15 timeout 90</td>
</tr>
<tr>
<td><strong>Step 8</strong></td>
<td>(Optional) Configures a generic bootstrap configuration.</td>
</tr>
<tr>
<td>Do one of the following: • discover {line line-type</td>
<td>controller controller-type</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-cns-conn)# discover interface serial</td>
</tr>
<tr>
<td><strong>Step 9</strong></td>
<td>Exits Cisco Networking Services connect configuration mode and returns to global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-cns-conn)# exit</td>
</tr>
<tr>
<td><strong>Step 10</strong></td>
<td>Starts the Cisco Networking Services configuration agent, connects to the Cisco Networking Services configuration engine, and initiates an initial configuration. You can use this command only before the system boots for the first time.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-cns-conn)# cns config initial {host-name</td>
</tr>
</tbody>
</table>
Configuring Advanced Cisco Networking Services Features

Perform this task to configure more advanced Cisco Networking Services features. After the Cisco Networking Services agents are operational, you can configure some other features. You can enable the Cisco Networking Services inventory agent—which is, send an inventory of the device’s line cards and modules to the Cisco Networking Services configuration engine—and enter Cisco Networking Services inventory mode.

Some other advanced features allow you to use the Software Developer’s Toolkit (SDK) to specify how Cisco Networking Services notifications should be sent or how to access MIB information. Two encapsulation methods can be used: either nongranular (SNMP) encapsulation or granular (XML) encapsulation.

SUMMARY STEPS

1. enable
2. configure terminal
3. cns mib-access encapsulation {snmp | xml[size bytes]}
4. cns notifications encapsulation {snmp | xml}
5. cns inventory
6. transport event
7. exit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>configure terminal</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# configure terminal</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>cns mib-access encapsulation {snmp</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# cns mib-access encapsulation snmp</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>cns notifications encapsulation {snmp</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# cns notifications encapsulation xml</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>cns inventory</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# cns inventory</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>transport event</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(cns-inv)# transport event</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>exit</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(cns-inv)# exit</td>
</tr>
</tbody>
</table>

### Troubleshooting Cisco Networking Services Agents

This section explains how to troubleshoot Cisco Networking Services agent issues.
The show commands created for the Cisco Networking Services image agent display information that is reset to zero after a successful reload of the device. Depending on the configuration of the image distribution process, the new image may not reload immediately. When a reload is not immediate or has failed, use the Cisco Networking Services image agent show commands to determine whether the image agent has connected to the image distribution server over HTTP or whether the image agent is receiving events from an application over the Cisco Networking Services Event Bus.

**SUMMARY STEPS**

1. enable
2. show cns image status
3. clear cns image status
4. show cns image connections
5. show cns image inventory
6. debug cns image [agent all connection error]
7. show cns event connections
8. show cns event subject [name]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables higher privilege levels, such as privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> show cns image status</td>
<td>(Optional) Displays information about the Cisco Networking Services image agent status.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device# show cns image status</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> clear cns image status</td>
<td>(Optional) Clears Cisco Networking Services image agent status statistics.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device# clear cns image status</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> show cns image connections</td>
<td>(Optional) Displays information about Cisco Networking Services image management server HTTP or HTTPS connections.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device# show cns image connections</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> show cns image inventory</td>
<td>(Optional) Displays inventory information about the Cisco Networking Services image agent.</td>
</tr>
<tr>
<td>Example:</td>
<td>• This command displays a dump of XML that would be sent out in response to an image agent inventory request message. The</td>
</tr>
<tr>
<td>Device# show cns image inventory</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td>XML output can be used to determine the information requested by an application.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> debug cns image [agent all] connection</td>
<td>error]</td>
</tr>
<tr>
<td>Example: Device# debug cns image all</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> show cns event connections</td>
<td>(Optional) Displays the status of the Cisco Networking Services event agent connection--such as whether it is connecting to the gateway, connected, or active--and to display the gateway used by the event agent and its IP address and port number.</td>
</tr>
<tr>
<td>Example: Device# show cns event connections</td>
<td></td>
</tr>
<tr>
<td><strong>Step 8</strong> show cns event subject [name]</td>
<td>(Optional) Displays a list of subjects of the Cisco Networking Services event agent that are subscribed to by applications.</td>
</tr>
<tr>
<td>Example: Device# show cns event subject subject1</td>
<td></td>
</tr>
</tbody>
</table>

**Examples**

In the following example, status information about the Cisco Networking Services image agent is displayed using the `show cns image status` privileged EXEC command:

```
Device# show cns image status
Last upgrade started at 11:45:02.000 UTC Mon May 6 2003
Last upgrade ended at 11:56:04.000 UTC Mon May 6 2003 status SUCCESS
Last successful upgrade ended at 11:56:04.000 UTC Mon May 6 2003
Last failed upgrade ended at 06:32:15.000 UTC Wed Apr 16 2003
Number of failed upgrades: 2
Number of successful upgrades: 6
messages received: 12
receive errors: 5
Transmit Status
TX Attempts:4
Successes:3 Failures 2
```

In the following example, information about the status of the Cisco Networking Services image management HTTP connections is displayed using the `show cns image connections` privileged EXEC command:

```
show cns image connections
CNS Image Agent: HTTP connections
Connection attempts 1
never connected:0  Abrupt disconnect:0
Last successful connection at 11:45:02.000 UTC Mon May 6 2003
```

In the following example, information about the Cisco Networking Services image agent inventory is displayed using the `show cns image inventory` privileged EXEC command:

```
show cns image inventory
Inventory Report
```
In the following example, debugging messages for all Cisco Networking Services image agent services are displayed using the `debug cns image` privileged EXEC command. The Cisco Networking Services image agent in this example is connecting to an image server over HTTP. After connecting, the image server asks for an inventory of the Cisco device.

```
Device# debug cns image all
All cns image debug flags are on
Device# cns image retrieve

May 7 06:11:42.175: CNS Image Agent: set EXEC lock
May 7 06:11:42.175: CNS Image Agent: received message from EXEC
May 7 06:11:42.175: CNS Image Agent: set session lock 1
May 7 06:11:42.175: CNS Image Agent: attempting to send to 
destination(http://10.1.36.8:8080/imgsrv/xgate):
<?xml version="1.0" encoding="UTF-8"?

xml version="1.0" encoding="UTF-8"?
cnsMessage version="1.0" senderCredentials userName
dvlpr-7200-6 /userName /senderCredentials
messageID dvlpr-7200-6_2 /messageID sessionControl imageSessionStart version="1.0"
/initiatorInfo trigger EXEC/trigger initiatorCredentials userName dvlpr-7200-6/userName
/initiatorCredentials /initiatorInfo /imageSessionStart /sessionControl /cnsMessage

May 7 06:11:42.175: CNS Image Agent: clear EXEC lock
May 7 06:11:42.175: CNS Image Agent: HTTP message sent url:http://10.1.36.8:8080/imgsrv/xgate
May 7 06:11:42.191: CNS Image Agent: response data alloc 4096 bytes
May 7 06:11:42.191: CNS Image Agent: HTTP req data freed
May 7 06:11:42.191: CNS Image Agent: receive message
xml version="1.0" encoding="UTF-8"?
cnsMessage Version="1.0"
senderCredentials
user Name myImageServer.cisco.com/userName
pass Word R0lGODlhcgGSALMAAAQCAEMmCZtuMFQxDS8b/passWord
/senderCredentials
messageID dvlpr-c2600-2-476456/messageID
request
replyTo
/serverReply http://10.1.36.8:8080/imgsrv/xgate /serverReply
/replyTo
/imageInventory
/inventoryItemList
all/
/inventoryItemList
/imageInventory
/request
/cnsMessage

The following example displays the IP address and port number of the primary and backup gateways:

```
Device# show cns event connections
The currently configured primary event gateway:
hostname is 10.1.1.1.
port number is 11011.
Event-Id is Internal test1
Keepalive setting:
none.
Connection status:
Connection Established.
The currently configured backup event gateway:
none.
The currently connected event gateway:
hostname is 10.1.1.1.
port number is 11011.

The following sample displays a list of subjects of the Cisco Networking Services event agent that are subscribed to by applications:

```
Device# show cns event subject
The list of subjects subscribed by applications.
cisco.cns.mibaccess:request
```
Configuration Examples for Cisco Networking Services

Example: Deploying the Cisco Networking Services Device

The following example shows an initial configuration on a remote device. The hostname of the remote device is the unique ID. The Cisco Networking Services configuration engine IP address is 172.28.129.22.

```plaintext
cns template connect template1
cli ip address negotiated
cli encapsulation ppp
cli ip directed-broadcast
cli no keepalive
cli no shutdown
exit
cns connect host1 retry-interval 30 retries 3
exit
hostname RemoteRouter
ip route 172.28.129.22 255.255.255.0 10.11.11.1
cns id Ethernet 0 ipaddress
cns config initial 10.1.1.1 no-persist
exit
```

Example: Using the Cisco Networking Services Zero Touch Solution

Configuring PPP on a Serial Interface

The following example shows the bootstrap configuration for configuring PPP on a serial interface:

```plaintext
cns template connect ppp-serial
cli ip address negotiated
cli encapsulation ppp
cli ip directed-broadcast
cli no keepalive
exit
cns template connect ip-route
cli ip route 10.0.0.0 0.0.0.0 ${next-hop}
exit
cns connect serial-ppp ping-interval 1 retries 1
discover interface serial
template ppp-serial
template ip-route
exit
hostname 26ML
cns config initial 10.1.1.1 no-persist inventory
```

Configuring PPP on an Asynchronous Interface

The following example shows the bootstrap configuration for configuring PPP on an asynchronous interface:

```plaintext
cns template connect async
cli modem InOut
    .
    .
```
exit
cns template connect async-interface
cli encapsulation ppp
cli ip unnumbered FastEthernet0/0
cli dialer rotary-group 0
exit
cns template connect ip-route
cli ip route 10.0.0.0 0.0.0.0 ${next-hop}
exit
cns connect async
discover line Async
template async
discover interface
template async-interface
template ip-route
exit
hostname async-example
cns config initial 10.1.1.1 no-persist inventory

Configuring HDLC on a Serial Interface
The following example shows the bootstrap configuration for configuring High-Level Data Link Control (HDLC) on a serial interface:

cns template connect hdlc-serial
cli ip address slarp retry 1
exit
cns template connect ip-route
cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
exit
cns connect async
discover interface serial
template hdlc-serial
template ip-route
exit
hostname host1
cns config initial 10.1.1.1 no-persist inventory

Configuring Aggregator Device Interfaces
The following examples show how to configure a standard serial interface and a serial interface bound to a controller on an aggregator device (also known as the DCE). In order for connectivity to be established, the aggregator device must have a point-to-point subinterface configured.

Standard Serial Interface

interface Serial0/1
   no ip address
   encapsulation frame-relay
   frame-relay intf-type dce
exit
interface Serial0/1.1 point-to-point
   10.0.0.0 255.255.255.0
   frame-relay interface-dlci 8

Serial Interface Bound to a Controller

ccontroller T1 0
   framing sf
   linecode ami
   channel-group 0 timeslots 1-24
exit
interface Serial0:0
   no ip address
   encapsulation frame-relay
Configuring IP over Frame Relay

The following example shows the bootstrap configuration for configuring IP over Frame Relay on a CPE device:

```
cns template connect setup-frame
c   cli encapsulation frame-relay
   exit

cns template connect ip-over-frame
c   cli frame-relay interface-dlci ${dlci}
c   cli ip address dynamic
   exit

cns template connect ip-route
   cli ip route 10.0.0.0 0.0.0.0 ${next-hop}
   exit

cns connect ip-over-frame
   discover interface Serial
   template setup-frame
   discover dci
   template ip-over-frame
   template ip-route
   exit

cns config initial 10.1.1.1
```

Configuring IP over Frame Relay over T1

The following example shows the bootstrap configuration for configuring IP over Frame Relay over T1 on a CPE device:

```
cns template connect setup-frame
c   cli encapsulation frame-relay
   exit

cns template connect ip-over-frame
c   cli frame-relay interface-dlci ${dlci}
c   cli ip address dynamic
   exit

cns template connect ip-route
   cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
   exit

cns template connect t1-controller
   cli framing esf
   cli linecode b8zs
   cli channel-group 0 timeslots 1-24 speed 56
   exit

cns connect ip-over-frame-over-t1
   discover controller T1
   template t1-controller
   discover interface
   template setup-frame
   discover dci
   template ip-over-frame
   template ip-route
   exit

cns config initial 10.1.1.1
```
## Additional References

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Cisco Networking Services commands: complete command syntax, command mode,</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
<tr>
<td>command history, defaults, usage guidelines, and examples.</td>
<td></td>
</tr>
<tr>
<td>Cisco Networking Services Configuration Engine</td>
<td>Cisco CNS Configuration Engine Administrator Guide, 1.3</td>
</tr>
</tbody>
</table>

### Standards and RFCs

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No new or modified standards/RFCs are supported by this feature, and</td>
</tr>
<tr>
<td></td>
<td>support for existing standards/RFCs has not been modified by this</td>
</tr>
<tr>
<td></td>
<td>feature.</td>
</tr>
<tr>
<td></td>
<td>--</td>
</tr>
</tbody>
</table>

### MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To locate and download MIBs for selected platforms, Cisco software</td>
</tr>
<tr>
<td></td>
<td>releases, and feature sets, use Cisco MIB Locator found at the following</td>
</tr>
<tr>
<td></td>
<td>URL:</td>
</tr>
<tr>
<td></td>
<td><a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

### Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>
Feature Information for Cisco Networking Services

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Networking Services</td>
<td>Cisco IOS XE Release 2.1 12.2(25)S 12.2(33) SRA 12.2(33)SB 12.2(33)SXI</td>
<td>The Cisco Networking Services feature is a collection of services that can provide remote event-driven configuring of Cisco IOS networking devices and remote execution of some CLI commands. The following commands were introduced or modified by this feature: clear cns config stats, clear cns counters, clear cns event stats, cli (cns), cns config cancel, cns config initial, cns config notify, cns config partial, cns config retrieve, cns connect, cns event, cns exec, cns id, cns template connect, cns trusted-server, debug cns config, debug cns exec, debug cns xml-parser, logging cns-events, show cns config stats, show cns event connections, show cns event stats, show cns event subject.</td>
</tr>
</tbody>
</table>
CHAPTER 2

CNS Configuration Agent

- Finding Feature Information, page 21
- Information About CNS Configuration Agent, page 21
- How to Configure CNS Configuration Agent, page 23
- Configuration Examples for CNS Configuration Agent, page 25
- Additional References, page 26
- Feature Information for CNS Configuration Agent, page 27

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About CNS Configuration Agent

Cisco Networking Services Configuration Agent

The Cisco Networking Services configuration agent is involved in the initial configuration and subsequent partial configurations on a Cisco device. To activate the Cisco Networking Services configuration agent, enter any of the `cns config` CLI commands.

Initial Cisco Networking Services Configuration

When a routing device first comes up, it connects to the configuration server component of the Cisco Networking Services configuration agent by establishing a TCP connection through the use of the `cns config`
initial command, a standard CLI command. The device issues a request and identifies itself by providing a unique configuration ID to the configuration server.

When the Cisco Networking Services web server receives a request for a configuration file, it invokes the Java servlet and executes the corresponding embedded code. The embedded code directs the Cisco Networking Services web server to access the directory server and file system to read the configuration reference for this device (configuration ID) and template. The Configuration Agent prepares an instantiated configuration file by substituting all the parameter values specified in the template with valid values for this device. The configuration server forwards the configuration file to the Cisco Networking Services web server for transmission to the routing device.

The Cisco Networking Services configuration agent accepts the configuration file from the Cisco Networking Services web server, performs XML parsing, checks syntax (optional), and loads the configuration file. The routing device reports the status of the configuration load as an event to which a network monitoring or workflow application can subscribe.


### Incremental Cisco Networking Services Configuration

Once the network is up and running, new services can be added using the Cisco Networking Services configuration agent. Incremental (partial) configurations can be sent to routing devices. The actual configuration can be sent as an event payload by way of the event gateway (push operation) or as a signal event that triggers the device to initiate a pull operation.

The routing device can check the syntax of the configuration before applying it. If the syntax is correct, the routing device applies the incremental configuration and publishes an event that signals success to the configuration server. If the device fails to apply the incremental configuration, it publishes an event that indicates an error.

Once the routing device has applied the incremental configuration, it can write the configuration to NVRAM or wait until signaled to do so.

### Synchronized Configuration

When a routing device receives a configuration, the device has the option to defer application of the configuration upon receipt of a write-signal event. The Cisco Networking Services Configuration Agent feature allows the device configuration to be synchronized with other dependent network activities.
# How to Configure CNS Configuration Agent

## Configuring the Cisco Networking Services Event and EXEC Agents

### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `cns config partial {host-name | ip-address} [encrypt] [port-number] [source interface name] [inventory]`
4. `logging cns-events [severity-level]`
5. `cns exec [encrypt] [port-number] [source {ip-address | interface-type-number}]`
6. `cns event {hostname | ip-address} [encrypt] [port-number] [backup] [failover-time seconds] [keepalive seconds retry-count] [source ip-address | interface-name][clock-timeout time] [reconnect-time time]`
7. `exit`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>enable</strong></td>
</tr>
<tr>
<td>Example:</td>
<td>Device&gt; enable</td>
</tr>
<tr>
<td></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>configure terminal</strong></td>
</tr>
<tr>
<td>Example:</td>
<td>Device# configure terminal</td>
</tr>
<tr>
<td></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>**cns config partial {host-name</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# cns config partial 172.28.129.22 80</td>
</tr>
<tr>
<td></td>
<td>(Optional) Starts the Cisco Networking Services configuration agent, which provides Cisco Networking Services configuration services to Cisco clients, and initiates an incremental (partial) configuration.</td>
</tr>
<tr>
<td></td>
<td>• Use the optional <code>port-number</code> argument to specify the port number for the configuration server. The default is 80.</td>
</tr>
<tr>
<td></td>
<td>• Use the optional <code>source</code> keyword and <code>ip-address</code> argument to specify the use of an IP address as the source for Cisco Networking Services configuration agent communications.</td>
</tr>
<tr>
<td></td>
<td>• Use the optional <code>inventory</code> keyword to send an inventory of the linecards and modules in the device to the Cisco Networking Services configuration engine as part of the HTTP request.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>The optional <code>encrypt</code> keyword is available only in images that support SSL.</td>
</tr>
</tbody>
</table>
### CNS Configuration Agent

#### Configuring the Cisco Networking Services Event and EXEC Agents

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 4** logging cns-events `[severity-level]` | (Optional) Enables XML-formatted system event message logging to be sent through the Cisco Networking Services event bus.  
  - Use the optional `severity-level` argument to specify the number or name of the desired severity level at which messages should be logged. The default is level 7 (debugging). |
| Example: Device(config)# logging cns-events 2 |                                                                                                                                                                                      |

| **Step 5** cns exec [encrypt] [port-number] [source {ip-address | interface-type-number}] | (Optional) Enables and configures the Cisco Networking Services EXEC agent, which provides Cisco Networking Services EXEC services to Cisco clients.  
  - Use the optional `port-number` argument to specify the port number for the EXEC server. The default is 80.  
  - Use the optional `source` keyword and `ip-address/interface-type-number` argument to specify the use of an IP address as the source for Cisco Networking Services EXEC agent communications. |
| Example: Device(config)# cns exec source 172.17.2.2 | Note: The optional `encrypt` keyword is available only in images that support SSL. |

| **Step 6** cns event {hostname | ip-address} [encrypt] [port-number] [backup] [failover-time seconds] [keepalive seconds retry-count] [source ip-address | interface-name] [clock-timeout time] [reconnect-time time] | Configures the Cisco Networking Services event gateway, which provides Cisco Networking Services event services to Cisco clients.  
  - The optional `encrypt` keyword is available only in images that support SSL.  
  - Use the optional `port-number` argument to specify the port number for the event server. The default is 11011 with no encryption and 11012 with encryption.  
  - Use the optional `backup` keyword to indicate that this is the backup gateway. Before configuring a backup gateway, ensure that a primary gateway is configured.  
  - Use the optional `failover-time` keyword and `seconds` argument to specify a time interval in seconds to wait for the primary gateway route after the route to the backup gateway is established.  
  - Use the optional `keepalive` keyword with the `seconds` and `retry-count` arguments to specify the keepalive timeout in seconds and the retry count.  
  - Use the optional `source` keyword and `ip-address/interface-name` argument to specify the use of an IP address as the source for Cisco Networking Services event agent communications.  
  - Use the optional `clock-timeout` keyword to specify the maximum time, in minutes, that the Cisco Networking Services event agent will wait for the clock to be set for transports (such as SSL) that require an accurate clock.  
  - Use the optional `reconnect-time` keyword to specify the configurable upper limit of the maximum retry timeout. |
| Example: Device(config)# cns event 172.28.129.22 source 172.22.2.1 | Note: Until the `cns event` command is entered, no transport connections to the Cisco Networking Services event bus are made and therefore no other Cisco Networking Services agents are operational. |
### Purpose

**Command or Action**

<table>
<thead>
<tr>
<th>Step 7</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>exit</td>
<td>Exits global configuration mode and returns to privileged EXEC mode.</td>
</tr>
</tbody>
</table>

**Example:**

```
Device(config)# exit
```

---

**Troubleshooting Tips**

- Use the `show cns event connections` command to check that the Cisco Networking Services event agent is connected to the Cisco Networking Services event gateway.

- Use the `show cns event subject` command to check that the image agent subject names are registered. Subject names for the Cisco Networking Services image agent begin with `cisco.mgmt.cns.image`.

---

### Configuration Examples for CNS Configuration Agent

#### Example: Enabling and Configuring Cisco Networking Services Agents

The following example shows various Cisco Networking Services agents being enabled and configured starting with the configuration agent being enabled with the `cns config partial` command to configure an incremental (partial) configuration on a remote device. The Cisco Networking Services configuration engine IP address is 172.28.129.22, and the port number is 80. The Cisco Networking Services exec agent is enabled with an IP address of 172.28.129.23, and the Cisco Networking Services event agent is enabled with an IP address of 172.28.129.24. Until the Cisco Networking Services event agent is enabled, no other Cisco Networking Services agents are operational.

```
cns config partial 172.28.129.22 80
```

```
cns exec 172.28.129.23 source 172.22.2.2
```

```
cns event 172.28.129.24 source 172.22.2.1
```

```
exit
```

In the following example, the Cisco Networking Services image agent parameters are configured using the CLI. An image ID is specified to use the IP address of the GigabitEthernet interface 0/1/1, a password is configured for the Cisco Networking Services image agent services, the Cisco Networking Services image upgrade retry interval is set to four minutes, and image management and status servers are configured.

```
cns id GigabitEthernet0/1/1 ipaddress image
```

```
cns image retry 240
```

```
cns image password abctext
```

```
```

In the following example, the Cisco Networking Services image agent is configured to use the Cisco Networking Services Event Bus. An image ID is specified as the hardware serial number of the networking device, the Cisco Networking Services event agent is enabled with a number of parameters, and the Cisco Networking Services event agent is enabled.

---

*Cisco Networking Services Configuration Guide, Cisco IOS Release 15S*
Services image agent is enabled without any keywords or options. The Cisco Networking Services image agent will listen for events on the Cisco Networking Services Event Bus.

cns id hardware-serial image
cns event 10.21.9.7 11011 keepalive 240 120 failover-time 5
cns image
cns image password abctext

Example: Retrieving a Cisco Networking Services Image from a Server

In the following example, the Cisco Networking Services image agent polls a file server using the cns image retrieve command. Assuming that the Cisco Networking Services image agent is already enabled, the file server and status server paths specified here will overwrite any existing image agent server and status configuration. The new file server will be polled and a new image, if it exists, will be downloaded to the networking device.


Additional References

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Cisco Networking Services commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples.</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
<tr>
<td>Cisco Networking Services Configuration Engine</td>
<td>Cisco CNS Configuration Engine Administrator Guide, 1.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards/RFCs are supported by this feature, and support for existing standards/RFCs has not been modified by this feature.</td>
<td>--</td>
</tr>
</tbody>
</table>
MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.</td>
<td>To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for CNS Configuration Agent

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
### Table 3: Feature Information for CNS Configuration Agent

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNS Configuration Agent</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The Cisco Networking Services Configuration Agent feature supports routing devices by providing the following:</td>
</tr>
<tr>
<td></td>
<td>12.0(18)ST</td>
<td>• Initial configurations</td>
</tr>
<tr>
<td></td>
<td>12.0(22)S</td>
<td>• Incremental (partial) configurations</td>
</tr>
<tr>
<td></td>
<td>12.2(2)T</td>
<td>• Synchronized configuration updates</td>
</tr>
<tr>
<td></td>
<td>12.2(8)T</td>
<td>The following commands were introduced or modified by this feature: <code>cns config cancel</code>, <code>cns config initial</code>, <code>cns config partial</code>, <code>cns config retrieve</code>, <code>cns password</code>, <code>debug cns config</code>, <code>debug cns xml-parser</code>, <code>show cns config outstanding</code>, <code>show cns config stats</code>, <code>show cns config status</code>.</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SRA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SXI</td>
<td></td>
</tr>
</tbody>
</table>
CNS Image Agent

- Finding Feature Information, page 29
- Prerequisites for CNS Image Agent, page 29
- Restrictions for CNS Image Agent, page 30
- Information About CNS Image Agent, page 30
- How to Configure CNS Image Agent, page 30
- Configuration Examples for CNS Image Agent, page 34
- Additional References, page 35
- Feature Information for CNS Image Agent, page 36

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for CNS Image Agent

- Determine where to store the Cisco images on a file server to make the image available to many other networking devices. If the Cisco Networking Services Event Bus is to be used to store and distribute the images, the Cisco Networking Services event agent must be configured.
- Set up a file server to enable the networking devices to download the new images. Protocols such as TFTP, HTTP, HTTPS, and rcp can be used.
- Determine how to handle error messages generated by Cisco Networking Services image agent operations. Error messages can be sent to the Cisco Networking Services Event Bus or an HTTP or HTTPS URL.
Restrictions for CNS Image Agent

During automated image loading operations you must try to prevent the Cisco device from losing connectivity with the file server that is providing the image. Image reloading is subject to memory issues and connection issues. Boot options must also be configured to allow the Cisco device to boot another image if the first image reload fails. For more details see the "Managing Configuration Files" module of the Cisco IOS Configuration Fundamentals Configuration Guide.

Information About CNS Image Agent

Cisco Networking Services Image Agent

Administrators maintaining large networks of Cisco devices need an automated mechanism to load image files onto large numbers of remote devices. Existing network management applications are useful to determine which images to run and how to manage images received from the Cisco online software center. Other image distribution solutions do not scale to cover thousands of devices and cannot distribute images to devices behind a firewall or using Network Address Translation (NAT). The Cisco Networking Services image agent enables the managed device to initiate a network connection and request an image download allowing devices using NAT, or behind firewalls, to access the image server.

The Cisco Networking Services image agent can be configured to use the Cisco Networking Services Event Bus. To use the Cisco Networking Services Event Bus, the Cisco Networking Services event agent must be enabled and connected to the Cisco Networking Services event gateway in the Cisco Networking Services Configuration Engine. The Cisco Networking Services image agent can also use an HTTP server that understands the Cisco Networking Services image agent protocol. Deployment of Cisco Networking Services image agent operations can use both the Cisco Networking Services Event Bus and an HTTP server.

How to Configure CNS Image Agent

Configuring the Cisco Networking Services Image Agent

Cisco Networking Services uses a unique identifier to identify an image agent associated with that Cisco device. Using the same process as Cisco Networking Services event and configuration agents, the configuration of the cns id command determines whether an IP address or MAC address of a specified interface, the hardware serial hardware number of the device, an arbitrary text string, or the hostname of the device is used as the image ID. By default, the system uses the hostname of the device.

The Cisco Networking Services image ID is sent in the content of the messages sent by the image agent and allows an application to know the unique image ID of the Cisco device that generated the message. A password can be configured and associated with the image ID in the image agent messages.

Before You Begin

Perform this task to configure Cisco Networking Services image agent parameters using CLI commands.
• To configure the Cisco Networking Services image agent to use HTTP or HTTP over SSL (HTTPS) to communicate with an image server, you need to know the URL for the image server and the URL to which status messages can be sent.

• If you are using HTTPS to communicate with the image server, you must set up security certificates to allow the server to be authenticated by the image agent when the connection is established.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. Do one of the following:
   - `cns id type number {ipaddress|mac-address} [event|image]`
   - `cns id {hardware-serial|hostname|string text} [event|image]`
4. `cns password password`
5. `cns image [server server-url[status status-url]]`
6. `cns image password image-password`
7. `cns image retry seconds`
8. `exit`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>configure terminal</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> Do one of the following:</td>
<td>Specifies a unique Cisco Networking Services ID and interface type and number from which to retrieve the unique ID. or Specifies a unique Cisco Networking Services ID assigned from the hardware serial number, device hostname, or an arbitrary text string. The following information applies to either version of the syntax.</td>
</tr>
</tbody>
</table>
### Purpose

**Command or Action**

**Example:**

Device(config)# cns id fastethernet 0/1 ipaddress image

**Example:**

Device(config)# cns id hardware-serial image

#### Step 4

**cns password password**

**Example:**

Device(config)# cns password password1

- Specifies a password for the Cisco Networking Services ID.
- You must configure the Cisco Networking Services password the first time a device is deployed, and the Cisco Networking Services password must be the same as the bootstrap password set on the Configuration Engine (CE).

#### Step 5

**cns image [server server-url[status status-url]]**

**Example:**


- Enables Cisco Networking Services image agent services and specifies the URL of the image distribution server.
- Use the optional **status** keyword and **status-url** argument to specify the URL of a web server to which error messages are written.
- If the **status** keyword and **status-url** argument are not specified, status messages are sent as events on the Cisco Networking Services Event Bus. To view the status messages on the Cisco Networking Services Event Bus, the Cisco Networking Services event agent must be configured.

#### Step 6

**cns image password image-password**

**Example:**

Device(config)# cns image password abctext

- (Optional) Specifies a password for Cisco Networking Services image agent services.
- If a password is configured, the password is included with the image ID in Cisco Networking Services image agent messages sent out by the image agent. The receiver of these messages can use this information to authenticate the sending device.

#### Step 7

**cns image retry seconds**

**Example:**

Device(config)# cns image retry 240

- (Optional) Specifies an image upgrade retry interval in seconds.
- The default interval is 60 seconds.

#### Step 8

**exit**

**Example:**

Device(config)# exit

- Exits global configuration mode and returns the device to privileged EXEC mode.

### What to Do Next

Proceed to the section to connect to the web server and download an image. If any of the commands in the task fail, proceed to the section to try to determine the problem.
## Retrieving a Cisco Networking Services Image from a Server

Perform this task to poll the image distribution server using HTTP or HTTPS.

### Troubleshooting Tips

- If the web server appears to be down, use the `ping` command to check connectivity.
- If using HTTP, use the `show ip http client all` command to display information about HTTP clients and connections.

### Before You Begin

This task assumes that you have already configured the Cisco Networking Services image agent using the tasks in the section.

### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `cns image retrieve [server server-url [status status-url]]`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> cns image retrieve [server server-url [status status-url]]</td>
<td>Contacts a Cisco Cisco Networking Services image distribution server and downloads a new image if a new image exists.</td>
</tr>
</tbody>
</table>
### Example: Enabling and Configuring Cisco Networking Services Agents

The following example shows various Cisco Networking Services agents being enabled and configured starting with the configuration agent being enabled with the `cns config partial` command to configure an incremental (partial) configuration on a remote device. The Cisco Networking Services configuration engine IP address is 172.28.129.22, and the port number is 80. The Cisco Networking Services exec agent is enabled with an IP address of 172.28.129.23, and the Cisco Networking Services event agent is enabled with an IP address of 172.28.129.24. Until the Cisco Networking Services event agent is enabled, no other Cisco Networking Services agents are operational.

```
cns config partial 172.28.129.22 80
```
```
cns exec 172.28.129.23 source 172.22.2.2
```
```
cns event 172.28.129.24 source 172.22.2.1
```
```
exit
```

In the following example, the Cisco Networking Services image agent parameters are configured using the CLI. An image ID is specified to use the IP address of the GigabitEthernet interface 0/1/1, a password is configured for the Cisco Networking Services image agent services, the Cisco Networking Services image upgrade retry interval is set to four minutes, and image management and status servers are configured.

```
cns id GigabitEthernet0/1/1 ipaddress image
cns image retry 240
```
```
cns image password abctext
```
```
```

In the following example, the Cisco Networking Services image agent is configured to use the Cisco Networking Services Event Bus. An image ID is specified as the hardware serial number of the networking device, the Cisco Networking Services event agent is enabled with a number of parameters, and the Cisco Networking Services image agent is enabled without any keywords or options. The Cisco Networking Services image agent will listen for events on the Cisco Networking Services Event Bus.

```
cns id hardware-serial image
cns event 10.21.9.7 11011 keepalive 240 120 failover-time 5
```
```
cns image
```
```
cns image password abctext
```
Example: Retrieving a Cisco Networking Services Image from a Server

In the following example, the Cisco Networking Services image agent polls a file server using the `cns image retrieve` command. Assuming that the Cisco Networking Services image agent is already enabled, the file server and status server paths specified here will overwrite any existing image agent server and status configuration. The new file server will be polled and a new image, if it exists, will be downloaded to the networking device.

```
```

Additional References

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Cisco Networking Services commands: complete command syntax, command mode,</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
<tr>
<td>command history, defaults, usage guidelines, and examples.</td>
<td></td>
</tr>
<tr>
<td>Cisco Networking Services Configuration Engine</td>
<td>Cisco CNS Configuration Engine Administrator Guide, 1.3</td>
</tr>
</tbody>
</table>

### Standards and RFCs

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards/RFCs are supported by this feature, and support</td>
<td>--</td>
</tr>
<tr>
<td>for existing standards/RFCs has not been modified by this feature.</td>
<td></td>
</tr>
</tbody>
</table>

### MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support</td>
<td>To locate and download MIBs for selected platforms, Cisco software</td>
</tr>
<tr>
<td>for existing MIBs has not been modified by this feature.</td>
<td>releases, and feature sets, use Cisco MIB Locator found at the following URL:</td>
</tr>
<tr>
<td></td>
<td><a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>
Feature Information for CNS Image Agent

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 4: Feature Information for Cisco Networking Services Image Agent

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Networking Services Image Agent</td>
<td>12.2(33)SEE</td>
<td>The Cisco Networking Services Image Agent feature is an infrastructure in Cisco IOS</td>
</tr>
<tr>
<td></td>
<td>12.3(1)</td>
<td>software to enable automated installation and activation of Cisco IOS images on Cisco</td>
</tr>
<tr>
<td></td>
<td>12.2(31)SB2</td>
<td>IOS networking devices. The following commands were introduced or modified by this</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SRB</td>
<td>feature: clear cns image connections, clear cns image status, cns id, cns image,</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SB</td>
<td>cns image password, cns image retrieve, cns image retry, debug cns image, show cns</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SXI</td>
<td>image connections, show cns image inventory, show cns image status.</td>
</tr>
</tbody>
</table>
CNS Event Agent

- Finding Feature Information, page 37
- Information About CNS Event Agent, page 37
- How to Configure CNS Event Agent, page 38
- Configuration Examples for CNS Event Agent, page 40
- Additional References, page 41
- Feature Information for CNS Event Agent, page 42

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About CNS Event Agent

Cisco Networking Services Event Agent

Although other Cisco Networking Services agents may be configured, no other Cisco Networking Services agents are operational until the cns event command is entered because the Cisco Networking Services event agent provides a transport connection to the Cisco Networking Services event bus for all other Cisco Networking Services agents. The other Cisco Networking Services agents use the connection to the Cisco Networking Services event bus to send and receive messages. The Cisco Networking Services event agent does not read or modify the messages.
# How to Configure CNS Event Agent

Configuring the Cisco Networking Services Event and EXEC Agents

## SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `cns config partial {host-name | ip-address} [encrypt] [port-number] [source interface name] [inventory]`
4. `logging cns-events [severity-level]`
5. `cns exec [encrypt] [port-number] [source {ip-address | interface-type-number}]`
6. `cns event {hostname | ip-address} [encrypt] [port-number] [backup] [failover-time seconds] [keepalive seconds retry-count] [source ip-address | interface-name][clock-timeout time] [reconnect-time time]`
7. `exit`

## DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device&gt; enable</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# configure terminal</td>
</tr>
<tr>
<td><strong>Step 3</strong> cns config partial {host-name</td>
<td>ip-address} [encrypt] [port-number] [source interface name] [inventory]</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# cns config partial 172.28.129.22 80</td>
</tr>
</tbody>
</table>

- Use the optional `port-number` argument to specify the port number for the configuration server. The default is 80.
- Use the optional `source` keyword and `ip-address` argument to specify the use of an IP address as the source for Cisco Networking Services configuration agent communications.
- Use the optional `inventory` keyword to send an inventory of the linecards and modules in the device to the Cisco Networking Services configuration engine as part of the HTTP request.

**Note** The optional `encrypt` keyword is available only in images that support SSL.
### CNS Event Agent

#### Configuring the Cisco Networking Services Event and EXEC Agents

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 4** logging cns-events [severity-level] | (Optional) Enables XML-formatted system event message logging to be sent through the Cisco Networking Services event bus.  
- Use the optional `severity-level` argument to specify the number or name of the desired severity level at which messages should be logged. The default is level 7 (debugging). |
| **Example:** Device(config)# logging cns-events 2 |
| **Step 5** cns exec [encrypt] [port-number] [source {ip-address | interface-type-number}] | (Optional) Enables and configures the Cisco Networking Services EXEC agent, which provides Cisco Networking Services EXEC services to Cisco clients.  
- Use the optional `port-number` argument to specify the port number for the EXEC server. The default is 80.  
- Use the optional `source` keyword and `ip-address/interface-type-number` argument to specify the use of an IP address as the source for Cisco Networking Services EXEC agent communications.  
**Note** The optional `encrypt` keyword is available only in images that support SSL. |
| **Example:** Device(config)# cns exec source 172.17.2.2 |
| **Step 6** cns event {hostname | ip-address} [encrypt] [port-number] [backup] [failover-time seconds] [backup] [keepalive seconds retry-count] [source ip-address | interface-name][clock-timeout time] [reconnect-time time] | Configures the Cisco Networking Services event gateway, which provides Cisco Networking Services event services to Cisco clients.  
- The optional `encrypt` keyword is available only in images that support SSL.  
- Use the optional `port-number` argument to specify the port number for the event server. The default is 11011 with no encryption and 11012 with encryption.  
- Use the optional `backup` keyword to indicate that this is the backup gateway. Before configuring a backup gateway, ensure that a primary gateway is configured.  
- Use the optional `failover-time` keyword and `seconds` argument to specify a time interval in seconds to wait for the primary gateway route after the route to the backup gateway is established.  
- Use the optional `keepalive` keyword with the `seconds` and `retry-count` arguments to specify the keepalive timeout in seconds and the retry count.  
- Use the optional `source` keyword and `ip-address/interface-name` argument to specify the use of an IP address as the source for Cisco Networking Services event agent communications.  
- Use the optional `clock-timeout` keyword to specify the maximum time, in minutes, that the Cisco Networking Services event agent will wait for the clock to be set for transports (such as SSL) that require an accurate clock.  
- Use the optional `reconnect-time` keyword to specify the configurable upper limit of the maximum retry timeout.  
**Note** Until the `cns event` command is entered, no transport connections to the Cisco Networking Services event bus are made and therefore no other Cisco Networking Services agents are operational.  
<p>| <strong>Example:</strong> Device(config)# cns event 172.28.129.22 source 172.22.2.1 |</p>
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 7</strong> exit</td>
<td>Exits global configuration mode and returns to privileged EXEC mode.</td>
</tr>
</tbody>
</table>

**Example:**

Device(config)# exit

**Troubleshooting Tips**

- Use the `show cns event connections` command to check that the Cisco Networking Services event agent is connected to the Cisco Networking Services event gateway.
- Use the `show cns event subject` command to check that the image agent subject names are registered. Subject names for the Cisco Networking Services image agent begin with `cisco.mgmt.cns.image`.

**Configuration Examples for CNS Event Agent**

**Example: Enabling and Configuring Cisco Networking Services Agents**

The following example shows various Cisco Networking Services agents being enabled and configured starting with the configuration agent being enabled with the `cns config partial` command to configure an incremental (partial) configuration on a remote device. The Cisco Networking Services configuration engine IP address is 172.28.129.22, and the port number is 80. The Cisco Networking Services exec agent is enabled with an IP address of 172.28.129.23, and the Cisco Networking Services event agent is enabled with an IP address of 172.28.129.24. Until the Cisco Networking Services event agent is enabled, no other Cisco Networking Services agents are operational.

```
cns config partial 172.28.129.22 80
  cns exec 172.28.129.23 source 172.22.2.2
  cns event 172.28.129.24 source 172.22.2.1
  exit
```

In the following example, the Cisco Networking Services image agent parameters are configured using the CLI. An image ID is specified to use the IP address of the GigabitEthernet interface 0/1/1, a password is configured for the Cisco Networking Services image agent services, the Cisco Networking Services image upgrade retry interval is set to four minutes, and image management and status servers are configured.

```
cns id GigabitEthernet0/1/1 ipaddress image
cns image retry 240
cns image password abctext
```

In the following example, the Cisco Networking Services image agent is configured to use the Cisco Networking Services Event Bus. An image ID is specified as the hardware serial number of the networking device, the Cisco Networking Services event agent is enabled with a number of parameters, and the Cisco Networking...
Services image agent is enabled without any keywords or options. The Cisco Networking Services image agent will listen for events on the Cisco Networking Services Event Bus.

cns id hardware-serial image
cns event 10.21.9.7 11011 keepalive 240 120 failover-time 5
cns image
cns image password abctext

Additional References

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Cisco Networking Services commands: complete command</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
<tr>
<td>syntax, command mode, command history, defaults, usage</td>
<td></td>
</tr>
<tr>
<td>guidelines, and examples.</td>
<td></td>
</tr>
<tr>
<td>Cisco Networking Services Configuration Engine</td>
<td>Cisco CNS Configuration Engine Administrator Guide, 1.3</td>
</tr>
</tbody>
</table>

Standards and RFCs

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards/RFCs are supported by this feature, and support</td>
<td>--</td>
</tr>
<tr>
<td>for existing standards/RFCs has not been modified by this feature.</td>
<td></td>
</tr>
</tbody>
</table>

MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support</td>
<td>To locate and download MIBs for selected platforms, Cisco software</td>
</tr>
<tr>
<td>for existing MIBs has not been modified by this feature.</td>
<td>releases, and feature sets, use Cisco MIB Locator found at the following</td>
</tr>
<tr>
<td></td>
<td>URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>
Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for CNS Event Agent

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Networking Services Event Agent</td>
<td>12.0(18)ST</td>
<td>The Cisco Networking Services Event Agent is part of the Cisco IOS infrastructure that allows Cisco IOS applications to publish and subscribe to events on a Cisco Networking Services Event Bus. Cisco Networking Services Event Agent works in conjunction with the Cisco Networking Services Configuration Agent feature. The following commands were introduced or modified by this feature: <em>cns event, show cns event connections, show cns event stats, show cns event subject.</em></td>
</tr>
<tr>
<td></td>
<td>12.0(22)S</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(2)T</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SRA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SXI</td>
<td></td>
</tr>
</tbody>
</table>
Cisco Networking Services Config Retrieve Enhancement with Retry and Interval

- Finding Feature Information, page 43
- Information About CNS Config Retrieve Enhancement with Retry and Interval, page 43
- How to Configure CNS Config Retrieve Enhancement with Retry and Interval, page 44
- Configuration Examples for CNS Config Retrieve Enhancement with Retry and Interval, page 45
- Additional References, page 46
- Feature Information for CNS Config Retrieve Enhancement with Retry and Interval, page 47

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About CNS Config Retrieve Enhancement with Retry and Interval

Cisco Networking Services Config Retrieve Enhancement with Retry and Interval

The Cisco Networking Services Config Retrieve Enhancement with Retry and Interval feature adds new functionality to the cns config retrieve command enabling you to specify the retry interval and an amount of time in seconds to wait before attempting to retrieve a configuration from a trusted server.
# How to Configure CNS Config Retrieve Enhancement with Retry and Interval

## Retrieving a Cisco Networking Services Configuration from a Server

Use this task to request the configuration of a device from a configuration server. Use the `cns trusted-server` command to specify which configuration server can be used (trusted).

**Before You Begin**

This task assumes that you have specified a trusted server.

### SUMMARY STEPS

1. `enable`  
2. `configure terminal`  
3. `cns config retrieve {host-name | ip-address} [encrypt] [port-number] [page page] [overwrite-startup] [retry retries interval seconds] [syntax-check] [no-persist] [source interface name] [status url] [event] [inventory]`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 1 | `enable` | Enables privileged EXEC mode.  
| | Example:  
| | Device> enable | • Enter your password if prompted. |
| Step 2 | `configure terminal` | Enters global configuration mode.  
| | Example:  
| | Device# configure terminal | |
| Step 3 | `cns config retrieve {host-name | ip-address} [encrypt] [port-number] [page page] [overwrite-startup] [retry retries interval seconds] [syntax-check] [no-persist] [source interface name] [status url] [event] [inventory]` | Allows the device to retrieve configuration data from a web server.  
| | Example:  
| | Device(config)# cns config retrieve server1 retry 5 interval 45 | • The `retry` keyword is a number in the range 1 to 100, and will prompt for an `interval` in the range 1 to 3600 seconds.  
| **Note** | Troubleshooting Tips  
| | If you need to stop the retrieval process, enter the Ctrl+Shift+6 key sequence. | |
Configuration Examples for CNS Config Retrieve Enhancement with Retry and Interval

Example: Retrieving a Cisco Networking Services Configuration from a Server

Retrieving Configuration Data from the Cisco Networking Services Trusted Server

The following example shows how to request a configuration from a trusted server at 10.1.1.1:

```cns trusted-server all 10.1.1.1
exit
```
```
cns config retrieve 10.1.1.1
```

The following example shows how to request a configuration from a trusted server at 10.1.1.1 and to configure a Cisco Networking Services configuration retrieve interval using the `cns config retrieve` command:

```cns trusted-server all 10.1.1.1
exit
```
```
cns config retrieve 10.1.1.1 retry 50 interval 1500
```

CNS Config Retrieve Attempt 1 out of 50 is in progress
Next `cns config retrieve retry` is in 1499 seconds (Ctrl-Shft-6 to abort this command).

00:26:40: %CNS-3-TRANSPORT: CNS_HTTP_CONNECTION_FAILED:10.1.1.1-Process= "CNS config retv", ipl= 0, pid= 43
00:26:40: %CNS-3-TRANSPORT: CNS_HTTP_CONNECTION_FAILED -Process= "CNS config retv", ipl= 0, pid= 43.....

```cns config retrieve 10.1.1.1
```

Applying the Retrieved Data to the Running Configuration File

The following example shows how to check and apply configuration data retrieved from the server to running configuration file only. The Cisco Networking Services Configuration Agent will attempt to retrieve configuration data at 30-second intervals until the attempt is successful, or is unsuccessful five times in these attempts.

```cns config retrieve 10.1.1.1 syntax-check no-persist retry 5 interval 30
```

Overwriting the Startup Configuration File with the Retrieved Data

The following example shows how to overwrite the startup configuration file with the configuration data retrieved from the server. The configuration data will not be applied to the running configuration.

```cns config retrieve 10.1.1.1 syntax-check no-persist retry 5 interval 30
```
```
cns config retrieve 10.1.1.1 overwrite-startup
```
## Additional References

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Cisco Networking Services commands: complete command syntax, command mode,</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
<tr>
<td>command history, defaults, usage guidelines, and examples.</td>
<td></td>
</tr>
<tr>
<td>Cisco Networking Services Configuration Engine</td>
<td>Cisco CNS Configuration Engine Administrator Guide, 1.3</td>
</tr>
</tbody>
</table>

### Standards and RFCs

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards/RFCs are supported by this feature, and support</td>
<td>--</td>
</tr>
<tr>
<td>for existing standards/RFCs has not been modified by this feature.</td>
<td></td>
</tr>
</tbody>
</table>

### MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support</td>
<td>To locate and download MIBs for selected platforms, Cisco software</td>
</tr>
<tr>
<td>for existing MIBs has not been modified by this feature.</td>
<td>releases, and feature sets, use Cisco MIB Locator found at the following</td>
</tr>
<tr>
<td></td>
<td>URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

### Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
<tr>
<td>download documentation, software, and tools. Use these resources to install</td>
<td></td>
</tr>
<tr>
<td>and configure the software and to troubleshoot and resolve technical</td>
<td></td>
</tr>
<tr>
<td>issues with Cisco products and technologies. Access to most tools on the</td>
<td></td>
</tr>
<tr>
<td>Cisco Support and Documentation website requires a Cisco.com user ID and</td>
<td></td>
</tr>
<tr>
<td>password.</td>
<td></td>
</tr>
</tbody>
</table>
Feature Information for CNS Config Retrieve Enhancement with Retry and Interval

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 6: Feature Information for Cisco Networking Services Config Retrieve Enhancement with Retry and Interval

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Networking Services Config Retrieve Enhancement with Retry and Interval</td>
<td>Cisco IOS XE Release 2.1 12.4(15)T 12.2(33)SRC 12.2(33)SB 12.2(50)SY</td>
<td>The Cisco Networking Services Config Retrieve Enhancement with Retry and Interval feature adds two options to the <code>cns config retrieve</code> command enabling you to specify an amount of time in seconds to wait before attempting to retrieve a configuration from a trusted server. The number of retries is restricted to 100 to prevent the configuration agent from indefinitely attempting to reach an unreachable server. Use the keyboard combination Ctrl-Shift-6 to abort the <code>cns config retrieve</code> command. The following command was modified by this feature: <code>cns config retrieve</code>.</td>
</tr>
</tbody>
</table>
Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About CNS Interactive CLI

Cisco Networking Services Interactive CLI

The Cisco Networking Services Interactive CLI feature provides a XML interface that allows you to send interactive commands to a device, such as commands that generate prompts for user input. A benefit of this feature is that interactive commands can be aborted before they have been fully processed. For example, for commands that generate a significant amount of output, the XML interface can be customized to limit the size of the output or the length of time allowed for the output to accumulate. The capability to use a programmable interface to abort a command before its normal termination (similar to manually aborting a command) can greatly increase the efficiency of diagnostic applications that might use this functionality. The new XML interface also allows for multiple commands to be processed in a single session. The response for each command is packaged together and sent in a single response event.
Additional References

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Cisco Networking Services commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
<tr>
<td>Cisco Networking Services Configuration Engine</td>
<td>Cisco CNS Configuration Engine Administrator Guide, 1.3</td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for CNS Interactive CLI

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
The Cisco Networking Services Interactive CLI feature introduces an XML interface that allows you to send interactive commands to a device, such as commands that generate prompts for user input.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Networking Services CLI</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The Cisco Networking Services Interactive CLI feature introduces an XML interface that allows you to send interactive commands to a device, such as commands that generate prompts for user input.</td>
</tr>
<tr>
<td></td>
<td>12.0(28)S</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(18)SXE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(18)SXF2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SRC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SXI</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 7

Cisco Networking Services Security Enhancement

The Cisco Networking Services Security Enhancement feature improves the security of Cisco Networking Services messages by authenticating sender credentials through the use of the SOAP message format.

- Finding Feature Information, page 53
- Information About Cisco Networking Services Security Enhancement, page 54
- How to Configure Cisco Networking Services Security Enhancement, page 55
- Configuration Examples for Cisco Networking Services Security Enhancement, page 56
- Additional References, page 56
- Feature Information for Cisco Networking Services Security Enhancement, page 57

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
Information About Cisco Networking Services Security Enhancement

Cisco Networking Services Security Enhancement

Cisco Networking Services messages can be configured to use the Cisco Networking Services SOAP message structure, in which the username and password are authenticated.

If authentication, authorization, and accounting (AAA) is configured, then Cisco Networking Services SOAP messages will be authenticated with AAA. If AAA is not configured, there will be no authentication. For backward compatibility, Cisco Networking Services will support the existing non-SOAP message format and will respond accordingly without security.

The `cns aaa authentication` command is required to turn on Cisco Networking Services Security Enhancement. This command determines whether the Cisco Networking Services messages are using AAA security or not. If the `cns aaa authentication` command is configured, then all incoming SOAP messages into the device are authenticated by AAA.

Cisco Networking Services Trusted Servers

Use the `cns trusted-server` command to specify a trusted server for an individual Cisco Networking Services agent or for all the Cisco Networking Services agents. To avoid security violations, you can build a list of trusted servers from which Cisco Networking Services agents can receive messages. An attempt to connect to a server not on the list will result in an error message being displayed.

Configure a Cisco Networking Services trusted server when a Cisco Networking Services agent will redirect its response to a server address that is not explicitly configured on the command line for the specific Cisco Networking Services agent. For example, the Cisco Networking Services EXEC agent may have one server configured but receive a message from the Cisco Networking Services event bus that overrides the configured server. The new server address has not been explicitly configured, so the new server address is not a trusted server. An error will be generated when the Cisco Networking Services exec agent tries to respond to this new server address unless the `cns trusted-server` command has been configured for the new server address.
How to Configure Cisco Networking Services Security Enhancement

Configuring Cisco Networking Services Trusted Servers

SUMMARY STEPS

1. enable
2. configure terminal
3. cns trusted-server {all-agents | config | event | exec | image} name
4. cns message format notification {version 1 | version 2}
5. cns aaa authentication authentication-method

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td></td>
<td>Device&gt; enable</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device# configure terminal</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 3</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cns trusted-server {all-agents</td>
<td>config</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config)# cns trusted-server event 10.19.2.5</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 4</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cns message format notification {version 1</td>
<td>version 2}</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td>Received messages which do not conform to the configured message format are rejected.</td>
</tr>
<tr>
<td></td>
<td>Device(config)# cns message format notification version 1</td>
<td>Use version 1 to configure the non-SOAP message format. Use version 2 for SOAP message format.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 5</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cns aaa authentication authentication-method</td>
<td>Enables Cisco Networking Services AAA options.</td>
</tr>
</tbody>
</table>
### Example: Configuring Cisco Networking Services Trusted Servers

```plaintext
enable
configure terminal
cns trusted-server event 10.19.2.5
cns message format notification version 2
cns aaa authentication method1
```

### Additional References

#### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Cisco Networking Services commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
<tr>
<td>Cisco Networking Services Configuration Engine</td>
<td>Cisco CNS Configuration Engine Administrator Guide, 1.3</td>
</tr>
</tbody>
</table>
Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for Cisco Networking Services Security Enhancement

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 8: Feature Information for Cisco Networking Services Security Enhancement

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Networking Services Security Enhancement</td>
<td>12.4(9)T</td>
<td>The Cisco Networking Services Security Enhancement feature improves the security of Cisco Networking Services messages by authenticating sender credentials through the use of the SOAP message format. The following commands were introduced or modified: cns aaa authentication, cns message format notification.</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SRA</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Command Scheduler

The EXEC CLI specified in a Command Scheduler policy list must neither generate a prompt nor can it be terminated using keystrokes. Command Scheduler is designed as a fully automated facility, and no manual intervention is permitted.
Information About Command Scheduler (Kron)

Command Scheduler

The Command Scheduler (KRON) Policy for System Startup feature enables support for the Command Scheduler upon system startup.

The Command Scheduler allows customers to schedule fully-qualified EXEC mode CLI commands to run once, at specified intervals, at specified calendar dates and times, or upon system startup. Originally designed to work with Cisco Networking Services commands, Command Scheduler now has a broader application. Using the Cisco Networking Services image agent feature, remote devices residing outside a firewall or using Network Address Translation (NAT) addresses can use Command Scheduler to launch CLI at intervals, to update the image running in the device.

Command Scheduler has two basic processes. A policy list is configured containing lines of fully-qualified EXEC CLI commands to be run at the same time or same interval. One or more policy lists are then scheduled to run after a specified interval of time, at a specified calendar date and time, or upon system startup. Each scheduled occurrence can be set to run either once only or on a recurring basis.

How to Configure Command Scheduler (Kron)

Configuring Command Scheduler Policy Lists and Occurrences

An occurrence for Command Scheduler is defined as a scheduled event. Policy lists are configured to run after a specified interval of time, at a specified calendar date and time, or upon system startup. Policy lists can be run once, as a one-time event, or as recurring events over time.

Command Scheduler occurrences can be scheduled before the associated policy list has been configured, but a warning will advise you to configure the policy list before it is scheduled to run.

Before You Begin

Perform this task to set up Command Scheduler policy lists of EXEC Cisco Networking Services commands and configure a Command Scheduler occurrence to specify the time or interval after which the Cisco Networking Services commands will run.

Command Scheduler Policy Lists

Policy lists consist of one or more lines of fully-qualified EXEC CLI commands. All commands in a policy list are executed when the policy list is run by Command Scheduler using the kron occurrence command. Use separate policy lists for CLI commands that are run at different times. No editor function is available, and the policy list is run in the order in which it was configured. To delete an entry, use the no form of the cli command followed by the appropriate EXEC command. If an existing policy list name is used, new entries are added to the end of the policy list. To view entries in a policy list, use the show running-config command. If a policy list is scheduled to run only once, it will not be displayed by the show running-config command after it has run.

Policy lists can be configured after the policy list has been scheduled, but each policy list must be configured before it is scheduled to run.

Command Scheduler Occurrences
The clock time must be set on the routing device before a Command Scheduler occurrence is scheduled to run. If the clock time is not set, a warning message will appear on the console screen after the **kron occurrence** command has been entered. Use the **clock** command or Network Time Protocol (NTP) to set the clock time.

The EXEC CLI to be run by Command Scheduler must be tested on the routing device to determine if it will run without generating a prompt or allowing execution interruption by keystrokes. Initial testing is important because Command Scheduler will delete the entire policy list if any CLI syntax fails. Removing the policy list ensures that any CLI dependencies will not generate more errors.

If you use the **conditional** keyword with the **kron policy-list** command, execution of the commands will stop when an error is encountered.

**Note**

- No more than 31 policy lists can be scheduled to run at the same time.
- If a one-time occurrence is scheduled, the occurrence will not be displayed by the **show running-config** command after the occurrence has run.

**SUMMARY STEPS**

1. **enable**
2. **configure terminal**
3. **kron policy-list** list-name [conditional]
4. **cli** command
5. **exit**
6. **kron occurrence** occurrence-name [user username] {in | [numdays:]numhours:]nummin|at hours:min|[month] day-of-month [day-of-week]} {oneshot| recurring | system-startup}
7. **policy-list** list-name
8. **exit**
9. **show kron schedule**

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device# configure terminal</td>
<td></td>
</tr>
</tbody>
</table>
### Command Scheduler Policy Lists and Occurrences

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 3** | **kron policy-list  list-name [conditional]**  
Example:  
Device(config)# kron policy-list  
cns-weekly  
| Specifies a name for a new or existing Command Scheduler policy list and enters kron-policy configuration mode.  
- If the *list-name* is new, a new policy list structure is created.  
- If the *list-name* exists, the existing policy list structure is accessed. The policy list is run in configured order with no editor function.  
- If the optional **conditional** keyword is used, execution of the commands stops when an error is encountered.  |
| **Step 4** | **cli  command**  
Example:  
Device(config-kron-policy)# cli  
cns image retrieve server  
https://10.19.2.3/cnsweek/ status  
https://10.19.2.3/cnsstatus/week/  
| Specifies the fully-qualified EXEC command and associated syntax to be added as an entry in the specified Command Scheduler policy list.  
- Each entry is added to the policy list in the order in which it is configured.  
- Repeat this step to add other EXEC CLI commands to a policy list to be executed at the same time or interval.  
**Note** EXEC commands that generate a prompt or can be terminated using keystrokes will cause an error.  |
| **Step 5** | **exit**  
Example:  
Device(config-kron-policy)# exit  
| Exits kron-policy configuration mode and returns the device to global configuration mode.  |
| **Step 6** | **kron occurrence  occurrence-name [user username]  
{in[[numdays:]numhours:]nummin| at hours:min[[month] day-of-month]  
[day-of-week]} [oneshot| recurring| system-startup]**  
Example:  
Device(config)# kron occurrence may  
user sales at 6:30 may 20 oneshot  
| Specifies a name and schedule for a new or existing Command Scheduler occurrence and enters kron-occurrence configuration mode.  
- Use the **in** keyword to specify a delta time interval with a timer that starts when this command is configured.  
- Use the **at** keyword to specify a calendar date and time.  
- Choose either the **oneshot** or **recurring** keyword to schedule Command Scheduler occurrence once or repeatedly. Add the optional **system-startup** keyword for the occurrence to be at system startup.  |
| **Step 7** | **policy-list  list-name**  
Example:  
Device(config-kron-occurrence)# policy-list sales-may  
| Specifies a Command Scheduler policy list.  
- Each entry is added to the occurrence list in the order in which it is configured.  
**Note** If the CLI commands in a policy list generate a prompt or can be terminated using keystrokes, an error will be generated and the policy list will be deleted.  |
### Command or Action | Purpose
---|---
**Step 8** | exit
**Example:**  
Device(config-kron-occurrence)# exit
Exits kron-occurrence configuration mode and returns the device to global configuration mode.
- Repeat this step to exit global configuration mode.

**Step 9** | show kron schedule
**Example:**  
Device# show kron schedule
(Optional) Displays the status and schedule information of Command Scheduler occurrences.

---

### Examples

In the following example, output information is displayed about the status and schedule of all configured Command Scheduler occurrences:

```
Device# show kron schedule
Kron Occurrence Schedule
  cns-weekly inactive, will run again in 7 days 01:02:33
  may inactive, will run once in 32 days 20:43:31 at 6:30 on May 20
```

### Troubleshooting Tips

Use the **debug kron** command in privileged EXEC mode to troubleshoot Command Scheduler command operations. Use any debugging command with caution because the volume of output generated can slow or stop the device's operations.

### Configuration Examples for Command Scheduler (Kron)

#### Example: Command Scheduler Policy Lists and Occurrences

In the following example, a Command Scheduler policy named cns-weekly is configured to run two sets of EXEC CLI involving Cisco Networking Services commands. The policy is then scheduled with two other policies to run every seven days, one hour and thirty minutes.

```
kron policy-list cns-weekly
cli cns config retrieve page /testconfig/config.asp no-persist
exit
kron occurrence week in 7:1:30 recurring
policy-list cns-weekly
policy-list itd-weekly
policy-list mkt-weekly
```
In the following example, a Command Scheduler policy named sales-may is configured to run a Cisco Networking Services command to retrieve a specified image from a remote server. The policy is then scheduled to run only once on May 20, at 6:30 a.m.

```
kron policy-list sales-may
cli cns image retrieve server 10.19.2.3 status 10.19.2.3
exit
kron occurrence may at 6:30 May 20 oneshot
policy-list sales-may
```

In the following example, a Command Scheduler policy named image-sunday is configured to run a Cisco Networking Services command to retrieve a specified image from a remote server. The policy is then scheduled to run every Sunday at 7:30 a.m.

```
kron policy-list image-sunday
cli cns image retrieve server 10.19.2.3 status 10.19.2.3
exit
kron occurrence sunday user sales at 7:30 sunday recurring
policy-list image-sunday
```

In the following example, a Command Scheduler policy named file-retrieval is configured to run a Cisco Networking Services command to retrieve a specific file from a remote server. The policy is then scheduled to run on system startup.

```
kron policy-list file-retrieval
cli cns image retrieve server 10.19.2.3 status 10.19.2.3
exit
kron occurrence system-startup
policy-list file-retrieval
```

### Additional References

**Related Documents**

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Cisco Networking Services commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples.</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
<tr>
<td>Cisco Networking Services Configuration Engine</td>
<td>Cisco CNS Configuration Engine Administrator Guide, 1.3</td>
</tr>
</tbody>
</table>

**Standards and RFCs**

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards/RFCs are supported by this feature, and support for existing standards/RFCs has not been modified by this feature.</td>
<td>--</td>
</tr>
</tbody>
</table>
MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.</td>
<td>To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for Command Scheduler (Kron)

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 9: Feature Information for Command Scheduler (Kron)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Scheduler (Kron)</td>
<td>Cisco IOS XE Release 2.1 12.3(1) 12.2(33)SRA 12.2(33)SRC 12.2(33)SB 12.2(33)SXI 12.2(50)SY</td>
<td>The Command Scheduler feature provides the ability to schedule some EXEC CLI commands to run at specific times or at specified intervals. The following commands were introduced or modified by this feature: cli, debug kron, kron occurrence, kron policy-list, policy-list, show kron schedule.</td>
</tr>
</tbody>
</table>
### Feature Information for Command Scheduler (Kron)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Scheduler (Kron) Policy for System Startup</td>
<td>12.2(33)SRC</td>
<td>The Command Scheduler (Kron) Policy for System Startup feature enables support for the Command Scheduler feature upon system startup.</td>
</tr>
<tr>
<td></td>
<td>12.2(50)SY</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.4(15)T</td>
<td></td>
</tr>
</tbody>
</table>
Network Configuration Protocol

The Network Configuration Protocol (NETCONF) defines a simple mechanism through which a network device can be managed, configuration data can be retrieved, and new configuration data can be uploaded and manipulated. NETCONF uses Extensible Markup Language (XML)-based data encoding for the configuration data and protocol messages.

- Finding Feature Information, page 67
- Prerequisites for NETCONF, page 67
- Information About NETCONF, page 68
- How to Configure NETCONF, page 68
- Configuration Examples for NETCONF, page 75
- Additional References for NETCONF, page 79
- Feature Information for NETCONF, page 80
- Glossary, page 80

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for NETCONF

A vty line must be available for each NETCONF session as specified by the netconf max-session command.
Information About NETCONF

NETCONF Notifications

NETCONF sends notifications of any configuration change over NETCONF. A notification is an event indicating that a configuration change has occurred. The change can be a new configuration, deleted configuration, or changed configuration. The notifications are sent at the end of a successful configuration operation as one message that shows the set of changes rather than showing individual messages for each line that is changed in the configuration.

How to Configure NETCONF

Configuring the NETCONF Network Manager Application

SUMMARY STEPS

1. Use the following CLI string to configure the NETCONF network manager application to invoke NETCONF as an SSH subsystem:

   Example:
   Unix Side: ssh -2 -s companyname@10.1.1.1 netconf

2. As soon as the NETCONF session is established, indicate the server capabilities by sending an XML document containing a <hello>:

   Example:
   
   ```xml
   <?xml version="1.0" encoding="UTF-8"?>
   <hello>
     <capabilities>
       <capability>
         urn:ietf:params:xml:ns:netconf:base:1.0
       </capability>
     </capabilities>
   </hello>
   ```

3. Use the following XML string to enable the NETCONF network manager application to send and receive NETCONF notifications:

4. Use the following XML string to stop the NETCONF network manager application from sending or receiving NETCONF notifications:

DETAILED STEPS

Step 1  Use the following CLI string to configure the NETCONF network manager application to invoke NETCONF as an SSH subsystem:

   Example:
   Unix Side: ssh -2 -s companyname@10.1.1.1 netconf

Step 2  As soon as the NETCONF session is established, indicate the server capabilities by sending an XML document containing a <hello>:

   Example:
   
   ```xml
   <?xml version="1.0" encoding="UTF-8"?>
   <hello>
     <capabilities>
       <capability>
         urn:ietf:params:xml:ns:netconf:base:1.0
       </capability>
     </capabilities>
   </hello>
   ```
The client also responds by sending an XML document containing a <hello>:

Example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<hello>
    <capabilities>
        <capability>
            urn:ietf:params:xml:ns:netconf:base:1.0
        </capability>
    </capabilities>
</hello>
```

Note Although the example shows the server sending a <hello> message followed by the message from the client, both sides send the message as soon as the NETCONF subsystem is initialized, perhaps simultaneously.

Tip All NETCONF requests must end with ]]>]]> which denotes an end to the request. Until the ]]>]]> sequence is sent, the device will not process the request.

See the "Example: Configuring NETCONF over SSHv2" section for a specific example.

**Step 3**

Use the following XML string to enable the NETCONF network manager application to send and receive NETCONF notifications:

Example:

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<rpc message-id="9.0"><notification-on/></rpc>
```

**Step 4**

Use the following XML string to stop the NETCONF network manager application from sending or receiving NETCONF notifications:

Example:

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<rpc message-id="9.13"><notification-off/></rpc>
```

Delivering NETCONF Payloads

Use the following XML string to deliver the NETCONF payload to the network manager application:

```xml
<?xml version="1.0" encoding="UTF-8"?>
    <!--The following elements define the cisco extensions for the content of the filter element in a <get-config> request. They allow the client to specify the format of the response and to select subsets of the entire configuration to be included.-->
    <xs:element name="config-format-text-block">
        <!--If this element appears in the filter, then the client is...-->
    </xs:element>
</xs:schema>
```
requesting that the response data be sent in config command block format.

<xs:annotation>
    <xs:documentation>
        When this element appears in the filter of a get-config request, the results are to be returned in E-DI XML format. The content of this element is treated as a filter.
    </xs:documentation>
</xs:annotation>

<![--These elements are used in the filter of a <get> to specify operational data to return.--]>
<xs:element name="oper-data-format-text-block">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="show" type="xs:string" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>

<![--When config-format-text format is specified, the following describes the content of the data element in the response--]>
<xs:element name="cli-config-data">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="cmd" type="xs:string" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>

<xs:element name="cli-config-data-block" type="xs:string">
    <xs:annotation>
        <xs:documentation>The content of this element is the device configuration as it would be sent to a terminal session. It contains embedded newline characters that must be preserved as they represent the boundaries between the individual command lines.</xs:documentation>
    </xs:annotation>
</xs:element>

<xs:element name="text-filter-spec">
    <xs:annotation>
        <xs:documentation>If this element is included in the config-format-text element, then the content is treated as if the string was appended to the "show running-config" command line.</xs:documentation>
    </xs:annotation>
</xs:element>
Formatting NETCONF Notifications

The NETCONF network manager application uses .xsd schema files to describe the format of the XML NETCONF notification messages that are sent between a NETCONF network manager application and a device running NETCONF over SSHv2 or BEEP. These files can be displayed in a browser or a schema reading tool. You can use these schemas to validate that the XML is correct. These schemas describe the format, not the content, of the data being exchanged.

NETCONF uses the <edit-config> function to load all of a specified configuration to a specified target configuration. When this new configuration is entered, the target configuration is not replaced. The target configuration is changed according to the data and requested operations of the requesting source.

The following are schemas for the NETCONF <edit-config> function in CLI, CLI block, and XML format.

**NETCONF <edit-config> Request: CLI Format**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <cli-config-data>
        <cmd>hostname test</cmd>
        <cmd>interface fastEthernet0/1</cmd>
        <cmd>ip address 192.168.1.1 255.255.255.0</cmd>
      </cli-config-data>
    </config>
  </edit-config>
</rpc>
```

**NETCONF <edit-config> Response: CLI Format**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="101" xmlns="urn:ietf:params:netconf:base:1.0">
  <ok/>
</rpc-reply>
```
NETCONF <edit-config> Request: CLI-Block Format

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="netconf.mini.edit.3">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <cli-config-data-block>
        hostname bob
        interface fastEthernet0/1
        ip address 192.168.1.1 255.255.255.0
      </cli-config-data-block>
    </config>
  </edit-config>
</rpc>
```

NETCONF <edit-config> Response: CLI-Block Format

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="netconf.mini.edit.3" xmlns="urn:ietf:params:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

The following are schemas for the NETCONF <get-config> function in CLI and CLI-block format.

NETCONF <get-config> Request: CLI Format

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter>
      <config-format-text-cmd>
        <text-filter-spec> | inc interface </text-filter-spec>
      </config-format-text-cmd>
    </filter>
  </get-config>
</rpc>
```

NETCONF <get-config> Response: CLI Format

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <cli-config-data>
      <cmd>interface FastEthernet0/1</cmd>
      <cmd>interface FastEthernet0/2</cmd>
    </cli-config-data>
  </data>
</rpc-reply>
```

NETCONF <get-config> Request: CLI-Block Format

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter>
      <config-format-text-block>
      </config-format-text-block>
    </filter>
  </get-config>
</rpc>
```
NETCONF <get-config> Response: CLI-Block Format

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <cli-config-data-block>
      <interface>
        FastEthernet0/1
        FastEthernet0/2
      </interface>
    </cli-config-data-block>
  </data>
</rpc-reply>
```

NETCONF uses the `<get>` function to retrieve configuration and device-state information. The NETCONF `<get>` format is the equivalent of a Cisco IOS `show` command. The `<filter>` parameter specifies the portion of the system configuration and device-state data to retrieve. If the `<filter>` parameter is empty, nothing is returned.

The following are schemas for the `<get>` function in CLI and CLI-block format.

### NETCONF <get> Request: CLI Format

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter>
      <config-format-text-cmd>
        include interface
      </text-filter-spec>
    </config-format-text-cmd>
    <oper-data-format-text-block>
      <exec>show interfaces</exec>
      <exec>show arp</exec>
    </oper-data-format-text-block>
  </filter>
</rpc>
```

### NETCONF <get> Response: CLI Format

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <cli-config-data>
      <cmd>interface Loopback0</cmd>
      <cmd>interface GigabitEthernet0/1</cmd>
      <cmd>interface GigabitEthernet0/2</cmd>
    </cli-config-data>
    <cli-oper-data-block>
      <item>
        <exec>show interfaces</exec>
        <response>
          <!-- output of "show interfaces" ---->
        </response>
      </item>
      <item>
        <exec>show arp</exec>
        <response>
          <!-- output of "show arp" ---->
        </response>
      </item>
    </cli-oper-data-block>
  </data>
</rpc-reply>
```
Monitoring and Maintaining NETCONF Sessions

Note

- A minimum of four concurrent NETCONF sessions must be configured.
- A maximum of 16 concurrent NETCONF sessions can be configured.
- NETCONF does not support SSHv1.
SUMMARY STEPS

1. enable
2. show netconf {counters | session | schema}
3. debug netconf {all | error}
4. clear netconf {counters | sessions}

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device&gt; enable</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Displays NETCONF information.</td>
</tr>
<tr>
<td>show netconf {counters</td>
<td>session</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# show netconf counters</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Enables debugging of NETCONF sessions.</td>
</tr>
<tr>
<td>debug netconf {all</td>
<td>error}</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# debug netconf error</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Clears NETCONF statistics counters and NETCONF sessions, and frees associated resources and locks.</td>
</tr>
<tr>
<td>clear netconf {counters</td>
<td>sessions}</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# clear netconf sessions</td>
</tr>
</tbody>
</table>

Configuration Examples for NETCONF

Example: Configuring the NETCONF Network Manager Application

The following example shows how to configure the NETCONF network manager application to invoke NETCONF as an SSH subsystem:

Unix Side: ssh-2 -s companyname@10.1.1.1 netconf
As soon as the NETCONF session is established, indicate the server capabilities by sending an XML document containing a <hello>:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<hello>
  <capabilities>
    <capability>
      urn:ietf:params:xml:ns:netconf:base:1.0
    </capability>
    <capability>
      urn:ietf:params:ns:netconf:capability:startup:1.0
    </capability>
  </capabilities>
  <session-id>4</session-id>
</hello>
```

The client also responds by sending an XML document containing a <hello>:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<hello>
  <capabilities>
    <capability>
      urn:ietf:params:xml:ns:netconf:base:1.0
    </capability>
  </capabilities>
</hello>
```

Use the following XML string to enable the NETCONF network manager application to send and receive NETCONF notifications:

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<rpc message-id="9.0"><notification-on/></rpc>
```

Use the following XML string to stop the NETCONF network manager application from sending or receiving NETCONF notifications:

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<rpc message-id="9.13"><notification-off/></rpc>
```

**Example: Monitoring NETCONF Sessions**

The following is sample output from the `show netconf counters` command:

```
Device# show netconf counters
NETCONF Counters
Connection Attempts:0: rejected:0 no-hello:0 success:0
Transactions
total:0, success:0, errors:0
detailed errors:
in-use 0   invalid-value 0    too-big 0
missing-attribute 0   bad-attribute 0    unknown-attribute 0
missing-element 0    bad-element 0     unknown-element 0
unknown-namespace 0    access-denied 0     lock-denied 0
resource-denied 0     rollback-failed 0   data-exists 0
data-missing 0  operation-not-supported 0     operation-failed 0
partial-operation 0
```

The following is sample output from the `show netconf session` command:

```
Device# show netconf session
(Current | max) sessions: 3 | 4
Operations received: 100    Operation errors: 99
```
The output of the `show netconf schema` command displays the element structure for a NETCONF request and the resulting reply. This schema can be used to construct proper NETCONF requests and parse the resulting replies. The nodes in the schema are defined in RFC 4741. The following is sample output from the `show netconf schema` command:

```
Device# show netconf schema
New Name Space 'urn:ietf:params:xml:ns:netconf:base:1.0'
<VirtualRootTag> [0, 1] required
  <rpc-reply> [0, 1] required
    <ok> [0, 1] required
      <data> [0, 1] required
        <rpc-error> [0, 1] required
          <error-type> [0, 1] required
          <error-tag> [0, 1] required
          <error-severity> [0, 1] required
          <error-app-tag> [0, 1] required
          <error-path> [0, 1] required
          <error-message> [0, 1] required
          <error-info> [0, 1] required
        </rpc-error>
        <bad-attribute> [0, 1] required
        <bad-element> [0, 1] required
        <ok-element> [0, 1] required
        <err-element> [0, 1] required
        <noop-element> [0, 1] required
        <bad-namespace> [0, 1] required
        <session-id> [0, 1] required
        <hello> [0, 1] required
        <capabilities> 1 required
        <capability> 1+ required
      </data>
    </rpc-reply>
  <close-session> [0, 1] required
  <commit> [0, 1] required
    <confirmed> [0, 1] required
    <confirm-timeout> [0, 1] required
  <copy-config> [0, 1] required
    <source> 1 required
      <config> [0, 1] required
        <cli-config-data> [0, 1] required
        <cmd> 1+ required
        <cli-config-data-block> [0, 1] required
        <xml-config-data> [0, 1] required
        <Device-Configuration> [0, 1] required
      </source>
    </copy-config>
  <default-operation> [0, 1] required
  <test-option> [0, 1] required
```

Network Configuration Protocol

Example: Monitoring NETCONF Sessions
Example: Monitoring NETCONF Sessions

```
<error-option> [0, 1] required
<config> 1 required
  <cli-config-data> [0, 1] required
  <cmd> 1+ required
  <cli-config-data-block> [0, 1] required
  <xml-config-data> [0, 1] required
    <Device-Configuration> [0, 1] required
    <> any subtree is allowed
  <get> [0, 1] required
    <filter> [0, 1] required
      <config-format-text-cmd> [0, 1] required
      <text-filter-spec> [0, 1] required
      <config-format-text-block> [0, 1] required
      <text-filter-spec> [0, 1] required
      <config-format-xml> [0, 1] required
      <oper-data-format-text-block> [0, 1] required
      <exec> [0, 1] required
      <show> [0, 1] required
      <oper-data-format-xml> [0, 1] required
      <exec> [0, 1] required
      <show> [0, 1] required
    <get-config> [0, 1] required
    <source> 1 required
      <config> [0, 1] required
      <cli-config-data> [0, 1] required
      <cmd> 1+ required
      <cli-config-data-block> [0, 1] required
      <xml-config-data> [0, 1] required
        <Device-Configuration> [0, 1] required
        <> any subtree is allowed
      <candidate> [0, 1] required
      <running> [0, 1] required
      <startup> [0, 1] required
      <url> [0, 1] required
      <filter> [0, 1] required
        <config-format-text-cmd> [0, 1] required
        <text-filter-spec> [0, 1] required
        <config-format-text-block> [0, 1] required
        <text-filter-spec> [0, 1] required
        <config-format-xml> [0, 1] required
        <oper-data-format-text-block> [0, 1] required
        <exec> [0, 1] required
        <show> [0, 1] required
        <oper-data-format-xml> [0, 1] required
        <exec> [0, 1] required
        <show> [0, 1] required
    <get-config> [0, 1] required
    <source> 1 required
      <config> [0, 1] required
      <cli-config-data> [0, 1] required
      <cmd> 1+ required
      <cli-config-data-block> [0, 1] required
      <xml-config-data> [0, 1] required
        <Device-Configuration> [0, 1] required
        <> any subtree is allowed
      <candidate> [0, 1] required
      <running> [0, 1] required
      <startup> [0, 1] required
      <url> [0, 1] required
      <filter> [0, 1] required
        <config-format-text-cmd> [0, 1] required
        <text-filter-spec> [0, 1] required
        <config-format-text-block> [0, 1] required
        <text-filter-spec> [0, 1] required
        <config-format-xml> [0, 1] required
        <oper-data-format-text-block> [0, 1] required
        <exec> [0, 1] required
        <show> [0, 1] required
        <oper-data-format-xml> [0, 1] required
        <exec> [0, 1] required
        <show> [0, 1] required
    <kill-session> [0, 1] required
      <session-id> [0, 1] required
      <lock> [0, 1] required
      <target> 1 required
        <candidate> [0, 1] required
        <running> [0, 1] required
        <startup> [0, 1] required
        <url> [0, 1] required
      <unlock> [0, 1] required
      <target> 1 required
        <candidate> [0, 1] required
        <running> [0, 1] required
        <startup> [0, 1] required
        <url> [0, 1] required
      <validate> [0, 1] required
      <source> 1 required
        <config> [0, 1] required
        <cli-config-data> [0, 1] required
        <cmd> 1+ required
        <cli-config-data-block> [0, 1] required
        <xml-config-data> [0, 1] required
        <Device-Configuration> [0, 1] required
        <> any subtree is allowed
        <candidate> [0, 1] required
        <running> [0, 1] required
        <startup> [0, 1] required
        <url> [0, 1] required
        <notification-on> [0, 1] required
        <notification-off> [0, 1] required
```
Additional References for NETCONF

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>NETCONF commands: complete command syntax, command mode, command history,</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
<tr>
<td>defaults, usage guidelines, and examples</td>
<td></td>
</tr>
<tr>
<td>Security and IP access lists commands: complete command syntax, command mode,</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
<tr>
<td>command history, defaults, usage guidelines, and examples</td>
<td></td>
</tr>
</tbody>
</table>

### Standards and RFCs

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 4251</td>
<td>The Secure Shell (SSH) Protocol Architecture</td>
</tr>
<tr>
<td>RFC 4252</td>
<td>The Secure Shell (SSH) Authentication Protocol</td>
</tr>
<tr>
<td>RFC 4741</td>
<td>NETCONF Configuration Protocol</td>
</tr>
<tr>
<td>RFC 4744</td>
<td>Using the NETCONF Protocol over the Blocks Extensible Exchange Protocol (BEEP)</td>
</tr>
</tbody>
</table>

### Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>
Feature Information for NETCONF

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 10: Feature Information for NETCONF

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>NETCONF</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The NETCONF protocol defines a simple mechanism through which a network device can be managed, configuration data can be retrieved, and new configuration data can be uploaded and manipulated. NETCONF uses Extensible Markup Language (XML)-based data encoding for the configuration data and protocol messages. The following commands were introduced or modified by this feature: clear netconf, debug netconf, show netconf.</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SRA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.2(33)SXI</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.4(9)T</td>
<td></td>
</tr>
<tr>
<td>NETCONF XML PI</td>
<td></td>
<td>The NETCONF protocol was enhanced, adding format attribute support for all Cisco IOS exec commands, including clear netconf, debug netconf, and show netconf.</td>
</tr>
</tbody>
</table>

Glossary


**NETCONF** — Network Configuration Protocol. A protocol that defines a simple mechanism through which a network device can be managed, configuration data can be retrieved, and new configuration data can be uploaded and manipulated.

**SASL** — Simple Authentication and Security Layer. An Internet standard method for adding authentication support to connection-based protocols. SASL can be used between a security appliance and a Lightweight Directory Access Protocol (LDAP) server to secure user authentication.
**SSHv2** — Secure Shell Version 2. SSH runs on top of a reliable transport layer and provides strong authentication and encryption capabilities. SSHv2 provides a means to securely access and securely execute commands on another computer over a network.

**TLS** — Transport Layer Security. An application-level protocol that provides for secure communication between a client and server by allowing mutual authentication, the use of hash for integrity, and encryption for privacy. TLS relies upon certificates, public keys, and private keys.

**XML** — Extensible Markup Language. A standard maintained by the World Wide Web Consortium (W3C) that defines a syntax that lets you create markup languages to specify information structures. Information structures define the type of information (for example, subscriber name or address), not how the information appears (bold, italic, and so on). External processes can manipulate these information structures and publish them in a variety of formats. XML allows you to define your own customized markup language.
You can use the Network Configuration Protocol (NETCONF) over Secure Shell Version 2 (SSHv2) feature to perform network configurations via the Cisco command-line interface (CLI) over an encrypted transport. The NETCONF Network Manager, which is the NETCONF client, must use Secure Shell Version 2 (SSHv2) as the network transport to the NETCONF server. Multiple NETCONF clients can connect to the NETCONF server.

- Finding Feature Information, page 83
- Prerequisites for NETCONF over SSHv2, page 83
- Restrictions for NETCONF over SSH, page 84
- Information About NETCONF over SSHv2, page 84
- How to Configure NETCONF over SSHv2, page 85
- Configuration Examples for NETCONF over SSHv2, page 91
- Additional References for NETCONF over SSHv2, page 93
- Feature Information for NETCONF over SSHv2, page 95

### Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

### Prerequisites for NETCONF over SSHv2

- NETCONF over SSHv2 requires that a vty line be available for each NETCONF session as specified in the `netconf max-session` command.
Restrictions for NETCONF over SSH

- Network Configuration Protocol (NETCONF) Secure Shell Version 2 (SSHv2) supports a maximum of 16 concurrent sessions.
- Only SSH version 2 is supported.

Information About NETCONF over SSHv2

NETCONF over SSHv2

To run the NETCONF over SSHv2 feature, the client (a Cisco device running Cisco software) establishes an SSH transport connection with the server (a NETCONF network manager). The following image shows a basic NETCONF over SSHv2 network configuration. The client and server exchange keys for security and password encryption. The user ID and password of the SSHv2 session running NETCONF are used for authorization and authentication purposes. The user privilege level is enforced and the client session may not have full access to the NETCONF operations if the privilege level is not high enough. If authentication, authorization, and accounting (AAA) is configured, the AAA service is used as if a user had established an SSH session directly to the device. Using the existing security configuration makes the transition to NETCONF almost seamless. Once the client has been successfully authenticated, the client invokes the SSH connection protocol and the SSH session is established. After the SSH session is established, the user or application invokes NETCONF as an SSH subsystem called "netconf."

Figure 1: NETCONF over SSHv2

Secure Shell Version 2

SSHv2 runs on top of a reliable transport layer and provides strong authentication and encryption capabilities. SSHv2 provides a means to securely access and securely execute commands on another computer over a network.
NETCONF does not support SSH version 1. The configuration for the SSH Version 2 server is similar to the configuration for SSH version 1. Use the `ip ssh version` command to specify which version of SSH that you want to configure. If you do not configure this command, SSH by default runs in compatibility mode; that is, both SSH version 1 and SSH version 2 connections are honored.

**Note**

SSH version 1 is a protocol that has never been defined in a standard. If you do not want your device to fall back to the undefined protocol (version 1), you should use the `ip ssh version` command and specify version 2.

Use the `ip ssh rsa keypair-name` command to enable an SSH connection using Rivest, Shamir, and Adelman (RSA) keys that you have configured. If you configure the `ip ssh rsa keypair-name` command with a key-pair name, SSH is enabled if the key pair exists, or SSH will be enabled if the key pair is generated later. If you use this command to enable SSH, you do not need to configure a hostname and a domain name.

## How to Configure NETCONF over SSHv2

### Enabling SSH Version 2 Using a Hostname and Domain Name

Perform this task to configure your device for SSH version 2 using a hostname and domain name. You may also configure SSH version 2 by using the RSA key pair configuration (see Enabling SSH Version 2 Using RSA Key Pairs, on page 86).

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `hostname hostname`
4. `ip domain-name name`
5. `crypto key generate rsa`
6. `ip ssh [timeout seconds | authentication-retries integer]`
7. `ip ssh version 2`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: <code>Device&gt; enable</code></td>
<td></td>
</tr>
<tr>
<td>• Enter your password if prompted.</td>
<td></td>
</tr>
</tbody>
</table>
## Enabling SSH Version 2 Using RSA Key Pairs

Perform this task to enable SSH version 2 without configuring a hostname or domain name. SSH version 2 will be enabled if the key pair that you configure already exists or if it is generated later. You may also configure SSH version 2 by using the hostname and domain name configuration. (See "Enabling SSH Version 2 Using a Hostname and Domain Name, on page 85.)

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>hostname hostname</td>
<td>Configures a hostname for your device.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config)# hostname host1</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>ip domain-name name</td>
<td>Configures a domain name for your device.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config)# ip domain-name domain1.com</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>crypto key generate rsa</td>
<td>Enables the SSH server for local and remote authentication.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config)# crypto key generate rsa</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>ip ssh [timeout seconds</td>
<td>(Optional) Configures SSH control variables on your device.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>authentication-retries integer]</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config)# ip ssh timeout 120</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>ip ssh version 2</td>
<td>Specifies the version of SSH to be run on your device.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config)# ip ssh version 2</td>
<td></td>
</tr>
</tbody>
</table>
### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip ssh rsa keypair-name keypair-name`
4. `crypto key generate rsa usage-keys label key-label modulus modulus-size`
5. `ip ssh [timeout seconds | authentication-retries integer]`
6. `ip ssh version 2`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| 1    | `enable`         | Enables privileged EXEC mode.  
|      |                  | * Enter your password if prompted. |
| 2    | `configure terminal` | Enters global configuration mode. |
| 3    | `ip ssh rsa keypair-name keypair-name` | Specifies which RSA keypair to use for SSH usage. |
|      |                  | **Note** A Cisco device can have many RSA key pairs. |
| 4    | `crypto key generate rsa usage-keys label key-label modulus modulus-size` | Enables the SSH server for local and remote authentication on the device.  
|      |                  | For SSH version 2, the modulus size must be at least 768 bits. |
|      |                  | **Note** To delete the RSA key pair, use the `crypto key zeroize rsa` command. After you have deleted the RSA command, you automatically disable the SSH server. |
| 5    | `ip ssh [timeout seconds | authentication-retries integer]` | Configures SSH control variables on your device. |
| 6    | `ip ssh version 2` | Specifies the version of SSH to be run on a device. |
Starting an Encrypted Session with a Remote Device

Perform this task to start an encrypted session with a remote networking device. (You do not have to enable your device. SSH can be run in disabled mode.)

From any UNIX or UNIX-like device, the following command is typically used to form an SSH session:

```
ssh -2 -s user@router.example.com netconf
```

### SUMMARY STEPS

1. Do one of the following:

   - `ssh [-v {1 | 2}] [-c {3des | aes128-cbc | aes192-cbc | aes256-cbc}] [-m {hmac-md5 | hmac-md5-96 | hmac-sha1 | hmac-sha1-96}] [-l userid] [-o numberofpasswordprompts n] [-p port-num] {ip-addr | hostname} [command]`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> Do one of the following:</td>
<td>Starts an encrypted session with a remote networking device.</td>
</tr>
<tr>
<td>`-v {1</td>
<td>2}] [-c {3des</td>
</tr>
</tbody>
</table>

**Example:**

```
Device# ssh -v 2 -c aes256-cbc -m hmac-sha1-96 -l user2 10.76.82.24
```

**Example:**

```
Device# ssh -v 2 -c aes256-cbc -m hmac-sha1-96 user2@10.76.82.24
```

### Troubleshooting Tips

The `ip ssh version` command can be used for troubleshooting your SSH configuration. By changing versions, you can determine which SSH version has a problem.
What to Do Next

For more information about the `ssh` command, see the Cisco IOS Security Command Reference.

Verifying the Status of the Secure Shell Connection

Perform this task to display the status of the SSH connection on your device.

Note

You can use the following `show` commands in user EXEC or privileged EXEC mode.

**SUMMARY STEPS**

1. `enable`
2. `show ssh`
3. `show ip ssh`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>(Optional) Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><code>enable</code></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device&gt; <code>enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Displays the status of SSH server connections.</td>
</tr>
<tr>
<td><code>show ssh</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device# <code>show ssh</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Displays the version and configuration data for SSH.</td>
</tr>
<tr>
<td><code>show ip ssh</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device# <code>show ip ssh</code></td>
<td></td>
</tr>
</tbody>
</table>

**Examples**

The following output from the `show ssh` command displays status about SSH version 2 connections.

```
Device# `show ssh`
Connection Version Mode Encryption  Hmac                   State
Username  Username
1         2.0     IN  aes128-cbc  hmac-md5     Session started lab
```
Enabling NETCONF over SSHv2

Perform this task to enable NETCONF over SSHv2.

**Before You Begin**

SSHv2 must be enabled.

---

**Note**

- A minimum of four concurrent NETCONF sessions must be configured.
- A maximum of 16 concurrent NETCONF sessions can be configured.
- NETCONF does not support SSHv1.

---

**SUMMARY STEPS**

1. enable
2. configure terminal
3. netconf ssh [acl access-list-number]
4. netconf lock-time seconds
5. netconf max-sessions session
6. netconf max-message size

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td>Enabling privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>Device&gt; enable</td>
<td>- Enter your password if prompted.</td>
</tr>
</tbody>
</table>

The following output from the `show ip ssh` command displays the version of SSH that is enabled, the authentication timeout values, and the number of authentication retries.

Device# show ip ssh
SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> netconf ssh [acl access-list-number]</td>
<td>Enables NETCONF over SSHv2.</td>
</tr>
<tr>
<td>Example: Device(config)# netconf ssh acl 1</td>
<td>- Optionally, you can configure an access control list for this NETCONF session.</td>
</tr>
<tr>
<td><strong>Step 4</strong> netconf lock-time seconds</td>
<td>(Optional) Specifies the maximum time, in seconds, a NETCONF configuration lock is in place without an intermediate operation.</td>
</tr>
<tr>
<td>Example: Device(config)# netconf lock-time 60</td>
<td>- The valid range is 1 to 300. The default value is 10 seconds.</td>
</tr>
<tr>
<td><strong>Step 5</strong> netconf max-sessions session</td>
<td>(Optional) Specifies the maximum number of concurrent NETCONF sessions allowed.</td>
</tr>
<tr>
<td>Example: Device(config)# netconf max-sessions 5</td>
<td>- The valid range is 4 to 16. The default value is 4.</td>
</tr>
<tr>
<td><strong>Step 6</strong> netconf max-message size</td>
<td>(Optional) Specifies the maximum size, in kilobytes (KB), for the messages received in a NETCONF session.</td>
</tr>
<tr>
<td>Example: Device(config)# netconf max-message 37283</td>
<td>- The valid range is 1 to 2147483. The default value is infinite.</td>
</tr>
<tr>
<td></td>
<td>- To set the maximum size to infinite, use the no netconf max-message command.</td>
</tr>
</tbody>
</table>

Configuration Examples for NETCONF over SSHv2

Example: Enabling SSHv2 Using a Hostname and Domain Name

```
configure terminal
hostname host1
ip domain-name example.com
crypto key generate rsa
ip ssh timeout 120
ip ssh version 2
```
Enabling Secure Shell Version 2 Using RSA Keys Example

The following example shows how to configure SSHv2 using RSA keys:

```
Device# configure terminal
Device(config)# ip ssh rsa keypair-name sshkeys
Device(config)# crypto key generate rsa usage-keys label sshkeys modulus 768
Device(config)# ip ssh timeout 120
Device(config)# ip ssh version 2
```

Starting an Encrypted Session with a Remote Device Example

The following example shows how to start an encrypted SSH session with a remote networking device, from any UNIX or UNIX-like device:

```
Device(config)# ssh -2 -s user@router.example.com netconf
```

Configuring NETCONF over SSHv2 Example

The following example shows how to configure NETCONF over SSHv2:

```
Device# configure terminal
Device(config)# netconf ssh acl 1
Device(config)# netconf lock-time 60
Device(config)# netconf max-sessions 5
Device(config)# netconf max-message 2345
Device# ssh-2 -s username@10.1.1.1 netconf
```

The following example shows how to get the configuration for loopback interface 113.

**SUMMARY STEPS**

1. First, send the "hello":
2. Next, send the get-config request:

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> First, send the &quot;hello&quot;:</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
</tbody>
</table>
| `<?xml version="1.0" encoding="UTF-8"?>
<hello><capabilities>
  <capability>urn:ietf:params:netconf:base:1.0</capability>
  <capability>urn:ietf:params:netconf:capability:writeable-running:1.0</capability>
  <capability>urn:ietf:params:netconf:capability:startup:1.0</capability>
  <capability>urn:ietf:params:netconf:capability:rollback-on-error:1.0</capability>
  <capability>urn:cisco:params:netconf:capability:pi-data-model:1.0</capability>
</capabilities>` | |
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| `<capability>urn:cisco:params:netconf:capability:notification:1.0</capability>`
|                   |         |
| `<capabilities>` |         |
| `<hello>`        |         |

**Step 2**

Next, send the get-config request:

**Example:**

```xml
```

The following output is shown on the device:

```xml
<?xml version="1.0" encoding="UTF-8"?><rpc-reply message-id="101" xmlns="urn:ietf:params:netconf:base:1.0"><data><cli-config-data><interface Loopback113><description test456><no ip address><load-interval 30><end></end></load-interval></end></no ip address></description><end></end></cli-config-data></data></rpc-reply>
```

### Additional References for NETCONF over SSHv2

**Related Documents**

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS Commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>NETCONF commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
</tbody>
</table>
### Related Topic

<table>
<thead>
<tr>
<th>Document Title</th>
<th>IP access lists commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Security commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples</td>
</tr>
<tr>
<td></td>
<td>IP access lists</td>
</tr>
<tr>
<td></td>
<td>Secure Shell and Secure Shell Version 2</td>
</tr>
</tbody>
</table>

### Standards and RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2246</td>
<td><em>The TLS Protocol Version 1.0</em></td>
</tr>
<tr>
<td>RFC 4251</td>
<td><em>The Secure Shell (SSH) Protocol Architecture</em></td>
</tr>
<tr>
<td>RFC 4252</td>
<td><em>The Secure Shell (SSH) Authentication Protocol</em></td>
</tr>
<tr>
<td>RFC 4741</td>
<td>NETCONF Configuration Protocol</td>
</tr>
<tr>
<td>RFC 4742</td>
<td>Using the NETCONF Configuration Protocol over Secure SHell (SSH)</td>
</tr>
</tbody>
</table>

### Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</td>
</tr>
</tbody>
</table>
Feature Information for NETCONF over SSHv2

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

**Table 11: Feature Information for NETCONF over SSHv2**

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>NETCONF over SSHv2</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The NETCONF over SSHv2 feature enables you to perform network configurations via the</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SB</td>
<td>Cisco command-line interface (CLI) over an encrypted transport.</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SRA</td>
<td>The following commands were introduced or modified by this feature: netconf lock-time,</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SXI</td>
<td>netconf max-message, netconf max-sessions netconf ssh.</td>
</tr>
<tr>
<td></td>
<td>12.4(9)T</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 11

NETCONF Access for Configurations over BEEP

You can use the Network Configuration Protocol (NETCONF) over Blocks Extensible Exchange Protocol (BEEP) feature to send notifications of any configuration change over NETCONF. A notification is an event indicating that a configuration change has happened. The change can be a new configuration, deleted configuration, or changed configuration. The notifications are sent at the end of a successful configuration operation as one message showing the set of changes, rather than individual messages for each line in the configuration that is changed.

BEEP can use the Simple Authentication and Security Layer (SASL) profile to provide simple and direct mapping to the existing security model. Alternatively, NETCONF over BEEP can use the transport layer security (TLS) to provide a strong encryption mechanism with either server authentication or server and client-side authentication.

- Finding Feature Information, page 97
- Prerequisites for NETCONF Access for Configurations over BEEP, page 98
- Restrictions for NETCONF Access for Configurations over BEEP, page 98
- Information About NETCONF Access for Configurations over BEEP, page 98
- How to Configure NETCONF Access for Configurations over BEEP, page 99
- Configuration Examples for NETCONF Access for Configurations over BEEP, page 104
- Additional References for NETCONF Access for Configurations over BEEP, page 104
- Feature Information for NETCONF Access for Configurations over BEEP, page 105

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
Prerequisites for NETCONF Access for Configurations over BEEP

NETCONF over BEEP listeners require Simple Authentication and Security layer (SASL) to be configured.

Restrictions for NETCONF Access for Configurations over BEEP

You must be running a crypto image in order to configure BEEP using transport layer security (TLS).

Information About NETCONF Access for Configurations over BEEP

NETCONF over BEEP Overview

The NETCONF Access for Configurations over BEEP feature allows you to enable BEEP as the transport protocol to use during NETCONF sessions. Using NETCONF over BEEP, you can configure either the NETCONF server or the NETCONF client to initiate a connection, thus supporting large networks of intermittently connected devices, and those devices that must reverse the management connection where there are firewalls and Network Address Translators (NATs).

BEEP is a generic application protocol framework for connection-oriented, asynchronous interactions. It is intended to provide the features that traditionally have been duplicated in various protocol implementations. BEEP typically runs on top of Transmission Control Protocol (TCP) and allows the exchange of messages. Unlike HTTP and similar protocols, either end of the connection can send a message at any time. BEEP also includes facilities for encryption and authentication and is highly extensible.

The BEEP protocol contains a framing mechanism that permits simultaneous and independent exchanges of messages between peers. These messages are usually structured using XML. All exchanges occur in the context of a binding to a well-defined aspect of the application, such as transport security, user authentication, or data exchange. This binding forms a channel; each channel has an associated profile that defines the syntax and semantics of the messages exchanged.

The BEEP session is mapped onto the NETCONF service. When a session is established, each BEEP peer advertises the profiles it supports. During the creation of a channel, the client (the BEEP initiator) supplies one or more proposed profiles for that channel. If the server (the BEEP listener) creates the channel, it selects one of the profiles and sends it in a reply. The server may also indicate that none of the profiles are acceptable, and decline creation of the channel.

BEEP allows multiple data exchange channels to be simultaneously in use.

Although BEEP is a peer-to-peer protocol, each peer is labeled according to the role it is performing at a given time. When a BEEP session is established, the peer that awaits new connections is the BEEP listener. The other peer, which establishes a connection to the listener, is the BEEP initiator. The BEEP peer that starts an exchange is the client, and the other BEEP peer is the server. Typically, a BEEP peer that acts in the server role also performs in the listening role. However, because BEEP is a peer-to-peer protocol, the BEEP peer that acts in the server role is not required to also perform in the listening role.
NETCONF over BEEP and SASL

The SASL is an Internet standard method for adding authentication support to connection-based protocols. SASL can be used between a security appliance and an Lightweight Directory Access Protocol (LDAP) server to secure user authentication.

BEEP listeners require SASL to be configured.

NETCONF over BEEP and TLS

The TLS is an application-level protocol that provides for secure communication between a client and server by allowing mutual authentication, the use of hash for integrity, and encryption for privacy. TLS relies upon certificates, public keys, and private keys.

Certificates are similar to digital ID cards. They prove the identity of the server to clients. Each certificate includes the name of the authority that issued it, the name of the entity to which the certificate was issued, the entity's public key, and time stamps that indicate the certificate's expiration date.

Public and private keys are the ciphers used to encrypt and decrypt information. Although the public key is shared, the private key is never given out. Each public-private key pair works together. Data encrypted with the public key can be decrypted only with the private key.

NETCONF over BEEP and Access Lists

You can optionally configure access lists for use with NETCONF over SSHv2 sessions. An access list is a sequential collection of permit and deny conditions that apply to IP addresses. The Cisco software tests addresses against the conditions in an access list one by one. The first match determines whether the software accepts or rejects the address. Because the software stops testing conditions after the first match, the order of the conditions is critical. If no conditions match, the software rejects the address.

The two main tasks involved in using access lists are as follows:

1. Creating an access list by specifying an access list number or name and access conditions.
2. Applying the access list to interfaces or terminal lines.

For more information about configuring access lists, see "IP Access List Overview" and "Creating an IP Access List and Applying It to an Interface" modules in Security Configuration Guide: Securing the Data Plane.

How to Configure NETCONF Access for Configurations over BEEP

Configuring an SASL Profile

To enable NETCONF over BEEP using SASL, you must first configure an SASL profile, which specifies which users are allowed access into the device.
### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `sasl profile profile-name`
4. `mechanism digest-md5`
5. `server user-name password password`
6. `exit`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | `enable` | Enables privileged EXEC mode.  
| | Example: Device> enable |  
| | | • Enter your password if prompted. |
| **Step 2** | `configure terminal` | Enters global configuration mode. |
| | Example: Device# configure terminal | |
| **Step 3** | `sasl profile profile-name` | Configures an SASL profile and enters SASL profile configuration mode. |
| | Example: Device(config)# sasl profile beep | |
| **Step 4** | `mechanism digest-md5` | Configures the SASL profile mechanism. |
| | Example: Device(config-SASL-profile)# mechanism digest-md5 | |
| **Step 5** | `server user-name password password` | Configures an SASL server. |
| | Example: Device(config-SASL-profile)# server user1 password password1 | |
| **Step 6** | `exit` | Exits global configuration mode and returns to privileged EXEC mode. |
| | Example: Device(config)# exit | |
Enabling NETCONF over BEEP

Before You Begin

- There must be at least as many vty lines configured as there are concurrent NETCONF sessions.
- If you configure NETCONF over BEEP using SASL, you must first configure an SASL profile.

Note

- A minimum of four concurrent NETCONF sessions must be configured.
- A maximum of 16 concurrent NETCONF sessions can be configured.

SUMMARY STEPS

1. enable
2. configure terminal
3. crypto key generate rsa general-keys
4. crypto pki trustpoint name
5. enrollment url url
6. subject-name name
7. revocation-check method1 [method2 [method3]]
8. exit
9. crypto pki authenticate name
10. crypto pki enroll name
11. netconf lock-time seconds
12. line vty line-number [ending-line-number]
13. netconf max-sessions session
14. netconf beep initiator {hostname | ip-address} port-number user sasl-user password sasl-password [encrypt trustpoint] [reconnect-time seconds]
15. netconf beep listener [port-number] [acl access-list-number] [sasl sasl-profile] [encrypt trustpoint]
16. exit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Step</td>
<td>Command or Action</td>
</tr>
<tr>
<td>------</td>
<td>-------------------</td>
</tr>
<tr>
<td>2</td>
<td><code>configure terminal</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device# configure terminal</td>
</tr>
<tr>
<td>3</td>
<td><code>crypto key generate rsa general-keys</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config)# crypto key generate rsa general-keys</td>
</tr>
<tr>
<td>4</td>
<td><code>crypto pki trustpoint name</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config)# crypto pki trustpoint my_trustpoint</td>
</tr>
<tr>
<td>5</td>
<td><code>enrollment url url</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(ca-trustpoint)# enrollment url <a href="http://10.2.3.3:80">http://10.2.3.3:80</a></td>
</tr>
<tr>
<td>6</td>
<td><code>subject-name name</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(ca-trustpoint)# subject-name CN=dns_name_of_host.com</td>
</tr>
<tr>
<td>7</td>
<td><code>revocation-check method1 [method2 [method3]]</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(ca-trustpoint)# revocation-check none</td>
</tr>
<tr>
<td>8</td>
<td><code>exit</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(ca-trustpoint)# exit</td>
</tr>
<tr>
<td>9</td>
<td><code>crypto pki authenticate name</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config)# crypto pki authenticate my_trustpoint</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 10</strong> crypto pki enroll name</td>
<td>Obtains the certificate or certificates for your router from CA.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# crypto pki enroll my_trustpoint</td>
<td></td>
</tr>
<tr>
<td><strong>Step 11</strong> netconf lock-time seconds</td>
<td>(Optional) Specifies the maximum time a NETCONF configuration lock is in place without an intermediate operation. The valid value range for the seconds argument is 1 to 300 seconds. The default value is 10 seconds.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# netconf lock-time 60</td>
<td></td>
</tr>
<tr>
<td><strong>Step 12</strong> line vty line-number [ending-line-number]</td>
<td>Identifies a specific virtual terminal line for remote console access. You must configure the same number of vty lines as maximum NETCONF sessions.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# line vty 0 15</td>
<td></td>
</tr>
<tr>
<td><strong>Step 13</strong> netconf max-sessions session</td>
<td>(Optional) Specifies the maximum number of concurrent NETCONF sessions allowed.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# netconf max-sessions 16</td>
<td></td>
</tr>
<tr>
<td><strong>Step 14</strong> netconf beep initiator {hostname</td>
<td>ip-address} port-number user sasl-user password sasl-password [encrypt trustpoint] [reconnect-time seconds]</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# netconf beep initiator host1 23 user user1 password password1 encrypt 23 reconnect-time 60</td>
<td></td>
</tr>
<tr>
<td><strong>Step 15</strong> netconf beep listener [port-number] [acl access-list-number] [sasl sasl-profile] [encrypt trustpoint]</td>
<td>(Optional) Specifies BEEP as the transport protocol for NETCONF and configures a peer as the BEEP listener. <strong>Note</strong> Perform this step to configure a NETCONF BEEP listener session. You can also optionally configure a BEEP initiator session.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# netconf beep listener 26 acl 101 sasl profile1 encrypt 25</td>
<td></td>
</tr>
<tr>
<td><strong>Step 16</strong> exit</td>
<td>Exits global configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# exit</td>
<td></td>
</tr>
</tbody>
</table>
Configuration Examples for NETCONF Access for Configurations over BEEP

Example: Enabling NETCONF over BEEP

Device# configure terminal
Device(config)# crypto key generate rsa general-keys
Device(config)# crypto pki trustpoint my_trustpoint
Device(ca-trustpoint)# enrollment url http://10.2.3.3:80
Device(ca-trustpoint)# subject-name CN=dns_name_of_host.com
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# crypto pki authenticate my_trustpoint
Device(ca-trustpoint)# crypto pki enroll my_trustpoint
Device(ca-trustpoint)# line vty 0 15
Device(ca-trustpoint)# exit
Device(config)# netconf lock-time 60
Device(config)# netconf max-sessions 16
Device(config)# netconf beep initiator host1 23 user my_user password my_password encrypt my_trustpoint reconnect-time 60
Device(config)# netconf beep listener 23 sasl user1 encrypt my_trustpoint

Additional References for NETCONF Access for Configurations over BEEP

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS Commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>NETCONF commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples</td>
<td>Cisco IOS Cisco Networking Services Command Reference</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2222</td>
<td>Simple Authentication and Security Layer (SASL)</td>
</tr>
<tr>
<td>RFC 3080</td>
<td>The Blocks Extensible Exchange Protocol Core</td>
</tr>
</tbody>
</table>
Feature Information for NETCONF Access for Configurations over BEEP

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 4741</td>
<td>NETCONF Configuration Protocol</td>
</tr>
<tr>
<td>RFC 4744</td>
<td>Using the NETCONF Protocol over the Blocks Extensible Exchange Protocol (BEEP)</td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>
The NETCONF over BEEP feature allows you to enable either the NETCONF server or the NETCONF client to initiate a connection, thus supporting large networks of intermittently connected devices and those devices that must reverse the management connection where there are firewalls and network address translators (NATs).

The following commands were introduced or modified by this feature: `netconf beep initiator`, `netconf beep listener`.

### Table 12: Feature Information for NETCONF Access for Configurations over BEEP

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| NETCONF Access for Configurations over BEEP | Cisco IOS XE Release 2.1  
12.2(33)SB  
12.2(33)SRB  
12.2(33)SXI  
12.4(9)T | The NETCONF over BEEP feature allows you to enable either the NETCONF server or the NETCONF client to initiate a connection, thus supporting large networks of intermittently connected devices and those devices that must reverse the management connection where there are firewalls and network address translators (NATs). The following commands were introduced or modified by this feature: `netconf beep initiator`, `netconf beep listener`. |