



# Configuring Cisco IOS Configuration Engine

---

- [Prerequisites for Configuring the Configuration Engine, on page 1](#)
- [Restrictions for Configuring the Configuration Engine, on page 1](#)
- [Information About Configuring the Configuration Engine, on page 1](#)
- [How to Configure the Configuration Engine, on page 5](#)
- [Monitoring CNS Configurations, on page 9](#)
- [Additional References, on page 10](#)
- [Feature Information for Cisco IOS Configuration Engine, on page 10](#)

## Prerequisites for Configuring the Configuration Engine

- Obtain the name of the configuration engine instance to which you are connecting.
- Because the CNS uses both the event bus and the configuration server to provide configurations to devices, you must define both ConfigID and Device ID for each configured device.
- All devices configured with the **cns config partial** global configuration command must access the event bus. The DeviceID, as originated on the device, must match the DeviceID of the corresponding device definition in the Cisco Configuration Engine. You must know the hostname of the event bus to which you are connecting.

## Restrictions for Configuring the Configuration Engine

- Within the scope of a single instance of the configuration server, no two configured devices can share the same value for ConfigID.
- Within the scope of a single instance of the event bus, no two configured devices can share the same value for DeviceID.

## Information About Configuring the Configuration Engine

This section provides information about configuring the configuration engine.

## Cisco Configuration Engine Software

The Cisco Configuration Engine is network management utility software that acts as a configuration service for automating the deployment and management of network devices and services. Each Cisco Configuration Engine manages a group of Cisco devices (switches and routers) and the services that they deliver, storing their configurations and delivering them as needed. The Cisco Configuration Engine automates initial configurations and configuration updates by generating device-specific configuration changes, sending them to the device, executing the configuration change, and logging the results.

The Cisco Configuration Engine supports standalone and server modes and has these Cisco Networking Services (CNS) components:

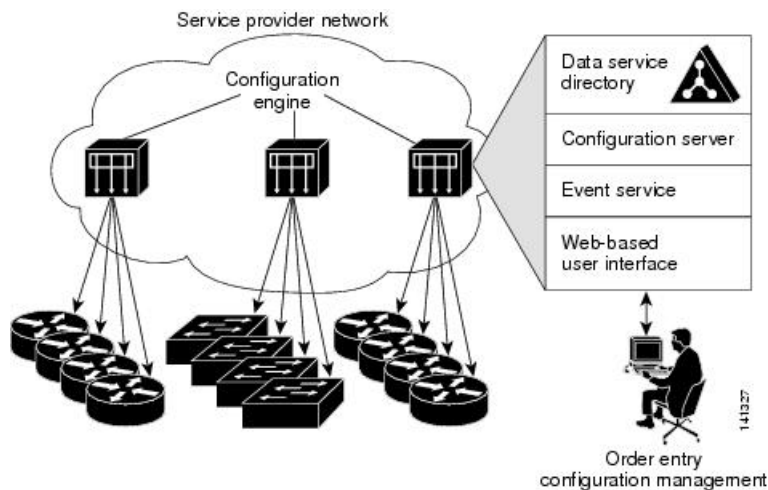
- Configuration service:
  - Web server
  - File manager
  - Namespace mapping server
- Event service (event gateway)
- Data service directory (data models and schema)



**Note** Support for Cisco Configuration Engine will be deprecated in future releases. Use the configuration described in [Cisco Plug and Play Feature Guide](#).

In standalone mode, the Cisco Configuration Engine supports an embedded directory service. In this mode, no external directory or other data store is required. In server mode, the Cisco Configuration Engine supports the use of a user-defined external directory.

**Figure 1: Cisco Configuration Engine Architectural Overview**



## Configuration Service

The Configuration Service is the core component of the Cisco Configuration Engine. It consists of a Configuration Server that works with Cisco IOS CNS agents on the device. The Configuration Service delivers device and service configurations to the device for initial configuration and mass reconfiguration by logical groups. Switches receive their initial configuration from the Configuration Service when they start up on the network for the first time.

The Configuration Service uses the CNS Event Service to send and receive configuration change events and to send success and failure notifications.

The Configuration Server is a web server that uses configuration templates and the device-specific configuration information stored in the embedded (standalone mode) or remote (server mode) directory.

Configuration templates are text files containing static configuration information in the form of CLI commands. In the templates, variables are specified by using Lightweight Directory Access Protocol (LDAP) URLs that reference the device-specific configuration information stored in a directory.

The Cisco IOS agent can perform a syntax check on received configuration files and publish events to show the success or failure of the syntax check. The configuration agent can either apply configurations immediately or delay the application until receipt of a synchronization event from the configuration server.

## Event Service

The Cisco Configuration Engine uses the Event Service for receipt and generation of configuration events. The Event Service consists of an event agent and an event gateway. The event agent is on the device and facilitates the communication between the devices and the event gateway on the Cisco Configuration Engine.

The Event Service is a highly capable publish-and-subscribe communication method. The Event Service uses subject-based addressing to send messages to their destinations. Subject-based addressing conventions define a simple, uniform namespace for messages and their destinations.

## NameSpace Mapper

The Cisco Configuration Engine includes the NameSpace Mapper (NSM) that provides a lookup service for managing logical groups of devices based on application, device or group ID, and event.

Cisco IOS devices recognize only event subject-names that match those configured in Cisco IOS software; for example, `cisco.cns.config.load`. You can use the namespace mapping service to designate events by using any desired naming convention. When you have populated your data store with your subject names, NSM changes your event subject-name strings to those known by Cisco IOS.

For a subscriber, when given a unique device ID and event, the namespace mapping service returns a set of events to which to subscribe. Similarly, for a publisher, when given a unique group ID, device ID, and event, the mapping service returns a set of events on which to publish.

## Cisco Networking Services IDs and Device Hostnames

The Cisco Configuration Engine assumes that a unique identifier is associated with each configured device. This unique identifier can take on multiple synonyms, where each synonym is unique within a particular namespace. The event service uses namespace content for subject-based addressing of messages.

The Cisco Configuration Engine intersects two namespaces, one for the event bus and the other for the configuration server. Within the scope of the configuration server namespace, the term *ConfigID* is the unique identifier for a device. Within the scope of the event bus namespace, the term *DeviceID* is the CNS unique identifier for a device.

## ConfigID

Each configured device has a unique ConfigID, which serves as the key into the Cisco Configuration Engine directory for the corresponding set of device CLI attributes. The ConfigID defined on the device must match the ConfigID for the corresponding device definition on the Cisco Configuration Engine.

The ConfigID is fixed at startup time and cannot be changed until the device restarts, even if the device hostname is reconfigured.

## DeviceID

Each configured device participating on the event bus has a unique DeviceID, which is analogous to the device source address so that the device can be targeted as a specific destination on the bus.

The origin of the DeviceID is defined by the Cisco IOS hostname of the device. However, the DeviceID variable and its usage reside within the event gateway adjacent to the device.

The logical Cisco IOS termination point on the event bus is embedded in the event gateway, which in turn functions as a proxy on behalf of the device. The event gateway represents the device and its corresponding DeviceID to the event bus.

The device declares its hostname to the event gateway immediately after the successful connection to the event gateway. The event gateway couples the DeviceID value to the Cisco IOS hostname each time this connection is established. The event gateway retains this DeviceID value for the duration of its connection to the device.

## Hostname and DeviceID

The DeviceID is fixed at the time of the connection to the event gateway and does not change even when the device hostname is reconfigured.

When changing the device hostname on the device, the only way to refresh the DeviceID is to break the connection between the device and the event gateway. For instructions on refreshing DeviceIDs, see "Related Topics."

When the connection is reestablished, the device sends its modified hostname to the event gateway. The event gateway redefines the DeviceID to the new value.



### Caution

When using the Cisco Configuration Engine user interface, you must first set the DeviceID field to the hostname value that the device acquires *after*, not *before*, and you must reinitialize the configuration for your Cisco IOS CNS agent. Otherwise, subsequent partial configuration command operations may malfunction.

## Hostname, DeviceID, and ConfigID

In standalone mode, when a hostname value is set for a device, the configuration server uses the hostname as the DeviceID when an event is sent on hostname. If the hostname has not been set, the event is sent on the `cn=<value>` of the device.

In server mode, the hostname is not used. In this mode, the unique DeviceID attribute is always used for sending an event on the bus. If this attribute is not set, you cannot update the device.

These and other associated attributes (tag value pairs) are set when you run **Setup** on the Cisco Configuration Engine.

## Automated CNS Configuration

To enable automated CNS configuration of the device, you must first complete the prerequisites listed in this topic. When you complete them, power on the device. At the **setup** prompt, do nothing; the device begins the initial configuration. When the full configuration file is loaded on your device, you do not need to do anything else.

For more information on what happens during initial configuration, see "Related Topics."

**Table 1: Prerequisites for Enabling Automatic Configuration**

Device	Required Configuration
Access device	Factory default (no configuration file)
Distribution device	<ul style="list-style-type: none"> <li>• IP helper address</li> <li>• Enable DHCP relay agent<sup>1</sup></li> <li>• IP routing (if used as default gateway)</li> </ul>
DHCP server	<ul style="list-style-type: none"> <li>• IP address assignment</li> <li>• TFTP server IP address</li> <li>• Path to bootstrap configuration file on the TFTP server</li> <li>• Default gateway IP address</li> </ul>
TFTP server	<ul style="list-style-type: none"> <li>• A bootstrap configuration file that includes the CNS configuration commands that enable the device to communicate with the Configuration Engine</li> <li>• The device configured to use either the device MAC address or the serial number (instead of the default hostname) to generate the ConfigID and EventID</li> <li>• The CNS event agent configured to push the configuration file to the device</li> </ul>
CNS Configuration Engine	One or more templates for each type of device, with the ConfigID of the device mapped to the template.

<sup>1</sup> A DHCP Relay is needed only when the DHCP Server is on a different subnet from the client.

## How to Configure the Configuration Engine

This section provides information about how to configure the configuration engine.

## Enabling the CNS Event Agent



**Note** You must enable the CNS event agent on the device before you enable the CNS configuration agent.

Follow these steps to enable the CNS event agent on the device.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Device> <b>enable</b>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>configure terminal</b> <b>Example:</b> Device# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 3</b>	<b>cns event</b> { <i>hostname</i>   <i>ip-address</i> } [ <i>port-number</i> ] [ <b>keepalive</b> <i>seconds</i> <i>retry-count</i> ] [ <b>failover-time</b> <i>seconds</i> ] [ <b>reconnect-time</b> <i>time</i> ]   <b>backup</b> ] <b>Example:</b> Device(config)# <b>cns event</b> 10.180.1.27 <b>keepalive</b> 120 10	Enables the event agent, and enters the gateway parameters. <ul style="list-style-type: none"> <li>• For {<i>hostname</i>   <i>ip-address</i>}, enter either the hostname or the IP address of the event gateway.</li> <li>• (Optional) For <i>port number</i>, enter the port number for the event gateway. The default port number is 11011.</li> <li>• (Optional) For <b>keepalive</b> <i>seconds</i>, enter how often the device sends keepalive messages. For <i>retry-count</i>, enter the number of unanswered keepalive messages that the device sends before the connection is terminated. The default for each is 0.</li> <li>• (Optional) For <b>failover-time</b> <i>seconds</i>, enter how long the device waits for the primary gateway route after the route to the backup gateway is established.</li> <li>• (Optional) For <b>reconnect-time</b> <i>time</i>, enter the maximum time interval that the device waits before trying to reconnect to the event gateway.</li> <li>• (Optional) Enter <b>backup</b> to show that this is the backup gateway. (If omitted, this is the primary gateway.)</li> </ul>

	Command or Action	Purpose
		<b>Note</b> Though visible in the command-line help string, the <b>encrypt</b> and the <b>clock-timeout</b> <i>time</i> keywords are not supported.
<b>Step 4</b>	<b>end</b> <b>Example:</b> Device (config)# <b>end</b>	Returns to privileged EXEC mode.
<b>Step 5</b>	<b>show running-config</b> <b>Example:</b> Device# <b>show running-config</b>	Verifies your entries.
<b>Step 6</b>	<b>copy running-config startup-config</b> <b>Example:</b> Device# <b>copy running-config startup-config</b>	(Optional) Saves your entries in the configuration file.

**What to do next**

To verify information about the event agent, use the **show cns event connections** command in privileged EXEC mode.

To disable the CNS event agent, use the **no cns event** { *ip-address* | *hostname* } global configuration command.

## Refreshing DeviceIDs

Follow these steps to refresh a DeviceID when changing the hostname on the device.

**Procedure**

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b> Device> <b>enable</b>	Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>
<b>Step 2</b>	<b>show cns config connections</b> <b>Example:</b> Device# <b>show cns config connections</b>	Displays whether the CNS event agent is connecting to the gateway, connected, or active, and the gateway used by the event agent, its IP address and port number.

	Command or Action	Purpose
<b>Step 3</b>	Make sure that the CNS event agent is properly connected to the event gateway.	Examine the output of <b>show cns config connections</b> for the following: <ul style="list-style-type: none"> <li>• Connection is active.</li> <li>• Connection is using the currently configured device hostname. The DeviceID will be refreshed to correspond to the new hostname configuration using these instructions.</li> </ul>
<b>Step 4</b>	<b>show cns event connections</b> <b>Example:</b>  Device# <b>show cns event connections</b>	Displays the event connection information for your device.
<b>Step 5</b>	Record from the output of Step 4 the information for the currently connected connection listed below. You will be using the IP address and port number in subsequent steps of these instructions.	
<b>Step 6</b>	<b>configure terminal</b> <b>Example:</b>  Device# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 7</b>	<b>no cns event ip-address port-number</b> <b>Example:</b>  Device(config)# <b>no cns event 172.28.129.22 2012</b>	Specifies the IP address and port number that you recorded in Step 5 in this command.  This command breaks the connection between the device and the event gateway. It is necessary to first break, then reestablish, this connection to refresh the DeviceID.
<b>Step 8</b>	<b>cns event ip-address port-number</b> <b>Example:</b>  Device(config)# <b>cns event 172.28.129.22 2012</b>	Specifies the IP address and port number that you recorded in Step 5 in this command.  This command reestablishes the connection between the device and the event gateway.
<b>Step 9</b>	<b>end</b> <b>Example:</b>  Device(config)# <b>end</b>	Returns to privileged EXEC mode.
<b>Step 10</b>	Make sure that you have reestablished the connection between the device and the event connection by examining the output from <b>show cns event connections</b> .	



	Command or Action	Purpose
<b>Step 11</b>	<b>show running-config</b> <b>Example:</b> Device# <code>show running-config</code>	Verifies your entries.
<b>Step 12</b>	<b>copy running-config startup-config</b> <b>Example:</b> Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

## Monitoring CNS Configurations

Table 2: CNS show Commands

Command	Purpose
<b>show cns config connections</b> Device# <code>show cns config connections</code>	Displays the status of the CNS Cisco IOS CNS agent connections.
<b>show cns config outstanding</b> Device# <code>show cns config outstanding</code>	Displays information about incremental (partial) CNS configurations that have started but are not yet completed.
<b>show cns config stats</b> Device# <code>show cns config stats</code>	Displays statistics about the Cisco IOS CNS agent.
<b>show cns event connections</b> Device# <code>show cns event connections</code>	Displays the status of the CNS event agent connections.
<b>show cns event gateway</b> Device# <code>show cns event gateway</code>	Displays the event gateway information for your device.
<b>show cns event stats</b> Device# <code>show cns event stats</code>	Displays statistics about the CNS event agent.
<b>show cns event subject</b> Device# <code>show cns event subject</code>	Displays a list of event agent subjects that are subscribed to by applications.

## Additional References

### Related Documents

Related Topic	Document Title
Configuration Engine Setup	<i>Cisco Configuration Engine Installation and Setup Guide, 1.5 for Linux</i> <a href="https://www.cisco.com/en/US/docs/net_mgmt/configuration_engine/1.5/installation_linux/guide/setup_1.html">https://www.cisco.com/en/US/docs/net_mgmt/configuration_engine/1.5/installation_linux/guide/setup_1.html</a>

## Feature Information for Cisco IOS Configuration Engine

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
	Cisco IOS Configuration Engine	The Cisco Configuration Engine is network management utility software that acts as a configuration service for automating the deployment and management of network devices and services.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.