



CHAPTER 5

Configuring Cisco IOS Configuration Engine

Finding Feature Information

Your software release may not support all the features documented in this chapter. For the latest feature information and caveats, see the release notes for your platform and software release.

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

Prerequisites for Configuring Cisco IOS Configuration Engine

Set the CNS DeviceID

- When using the Cisco Configuration Engine user interface, you must first set the DeviceID field to the hostname value that the switch acquires after, not before, you use the **cns config initial** global configuration command at the switch. Otherwise, subsequent **cns config partial** global configuration command operations malfunction.

Enable Automated CNS Configuration

- To enable automated CNS configuration of the switch, you must first complete the prerequisites in [Table 5-1](#). When you complete them, power on the switch. At the **setup** prompt, you do not need to enter a command. The switch begins the initial configuration as described in the “[Initial Configuration](#)” section on [page 5-5](#). When the full configuration file is loaded on your switch, you do not need to do anything else.

Table 5-1 Prerequisites for Enabling Automatic Configuration

Device	Required Configuration
Access switch	Factory default (no configuration file)
Distribution switch	<ul style="list-style-type: none">• IP helper address• Enable DHCP relay agent• IP routing (if used as default gateway)

Table 5-1 Prerequisites for Enabling Automatic Configuration (continued)

Device	Required Configuration
DHCP server	<ul style="list-style-type: none"> • IP address assignment • TFTP server IP address • Path to bootstrap configuration file on the TFTP server • Default gateway IP address
TFTP server	<ul style="list-style-type: none"> • A bootstrap configuration file that includes the CNS configuration commands that enable the switch to communicate with the Configuration Engine • The switch configured to use either the switch MAC address or the serial number (instead of the default hostname) to generate the ConfigID and EventID • The CNS event agent configured to push the configuration file to the switch
CNS Configuration Engine	One or more templates for each type of device, with the ConfigID of the device mapped to the template

Information About Configuring Cisco IOS Configuration Engine

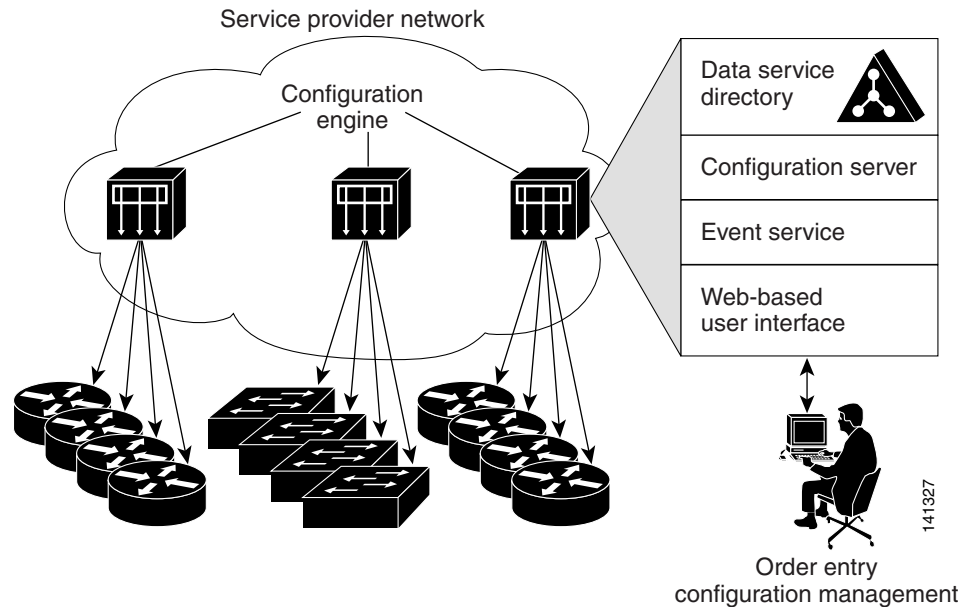
Cisco Configuration Engine is network management software that acts as a configuration service for automating the deployment and management of network devices and services (see [Figure 5-1](#)). Each Cisco Configuration Engine service manages a group of Cisco devices (switches and routers) and the services that they deliver, storing their configurations and delivering them as needed. 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.

Cisco Configuration Engine supports standalone and server modes and has these CNS components:

- Configuration service (web server, file manager, and namespace mapping server)
- Event service (event gateway)
- Data service directory (data models and schema)

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

Figure 5-1 Configuration Engine Architectural Overview



141327

Configuration Service

Configuration Service is the core component of Cisco Configuration Engine. It consists of a configuration server that works with Cisco IOS CNS agents on the switch. Configuration Service delivers device and service configurations to the switch 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.

Configuration Service uses 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 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

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

Event Service is a highly capable publish-and-subscribe communication method. 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

Configuration Engine includes NameSpace Mapper (NSM), which 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.

CNS IDs and Device Hostnames

Configuration Engine assumes that a unique identifier is associated with each configured switch. 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.

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.

Because Configuration Engine 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 switch.

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

ConfigID

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

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

DeviceID

Each configured switch participating on the event bus has a unique DeviceID, which is analogous to the switch source address so that the switch can be targeted as a specific destination on the bus. All switches configured with the **cns config partial** global configuration command must access the event bus. Therefore, the DeviceID, as originated on the switch, must match the DeviceID of the corresponding switch definition in Configuration Engine.

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

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 switch. The event gateway represents the switch and its corresponding DeviceID to the event bus.

The switch 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 caches this DeviceID value for the duration of its connection to the switch.

Hostname and DeviceID Interaction

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

When changing the switch hostname on the switch, the only way to refresh the DeviceID is to break the connection between the switch and the event gateway. Enter the **no cns event** global configuration command followed by the **cns event** global configuration command.

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

Using Hostname, DeviceID, and ConfigID

In standalone mode, when a hostname value is set for a switch, 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 switch.

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

Cisco IOS Agents

The CNS event agent feature allows the switch to publish and subscribe to events on the event bus and works with the Cisco IOS agent.

Initial Configuration

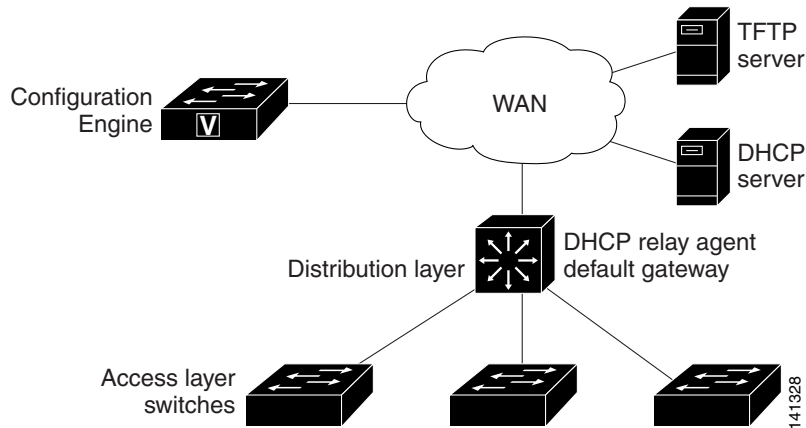
When the switch first comes up, it attempts to get an IP address by broadcasting a DHCP request on the network. Assuming there is no DHCP server on the subnet, the distribution switch acts as a DHCP relay agent and forwards the request to the DHCP server. Upon receiving the request, the DHCP server assigns an IP address to the new switch and includes the TFTP server IP address, the path to the bootstrap configuration file, and the default gateway IP address in a unicast reply to the DHCP relay agent. The DHCP relay agent forwards the reply to the switch.

The switch automatically configures the assigned IP address on interface VLAN 1 (the default) and downloads the bootstrap configuration file from the TFTP server. Upon successful download of the bootstrap configuration file, the switch loads the file in its running configuration.

The Cisco IOS agents initiate communication with Configuration Engine by using the appropriate ConfigID and EventID. Configuration Engine maps the ConfigID to a template and downloads the full configuration file to the switch.

Figure 5-2 shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

Figure 5-2 Initial Configuration Overview



Incremental (Partial) Configuration

After the network is running, new services can be added by using the Cisco IOS agent. Incremental (partial) configurations can be sent to the switch. 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 switch to initiate a pull operation.

The switch can check the syntax of the configuration before applying it. If the syntax is correct, the switch applies the incremental configuration and publishes an event that signals success to the configuration server. If the switch does not apply the incremental configuration, it publishes an event showing an error status. When the switch has applied the incremental configuration, it can write it to NVRAM or wait until signaled to do so.

Synchronized Configuration

When the switch receives a configuration, it can defer application of the configuration upon receipt of a write-signal event. The write-signal event tells the switch not to save the updated configuration into its NVRAM. The switch uses the updated configuration as its running configuration. This ensures that the switch configuration is synchronized with other network activities before saving the configuration in NVRAM for use at the next reboot.

How to Configure Cisco IOS Configuration Engine

Configuring Cisco IOS Agents

CNS Event Agent and Cisco IOS CNS Agent embedded in the Cisco IOS software on the switch allows the switch to be connected and automatically configured. Both agents must be enabled and the CNS configuration can be **initial** or **partial**. The partial configuration allows you to use Configuration Engine to remotely send incremental configuration to the switch.

Enabling CNS Event Agent

Before You Begin


You must enable CNS Event Agent on the switch before you enable Cisco IOS CNS Agent.

	Command	Purpose
Step 1	configure terminal	Enters global configuration mode.
Step 2	cns event { <i>hostname</i> <i>ip-address</i> } [<i>port-number</i>] [backup] [failover-time <i>seconds</i>] [keepalive <i>seconds</i>] <i>retry-count</i>] [reconnect <i>time</i>] [source <i>ip-address</i>]	<p>Enables the event agent, and enters the gateway parameters.</p> <ul style="list-style-type: none"> • {<i>hostname</i> <i>ip-address</i>}—Enters either the hostname or the IP address of the event gateway. • (Optional) <i>port number</i>—Enters the port number for the event gateway. The default port number is 11011. • (Optional) backup—Shows that this is the backup gateway. (If omitted, this is the primary gateway.) • (Optional) failover-time <i>seconds</i>—Enters how long the switch waits for the primary gateway route after the route to the backup gateway is established. • (Optional) keepalive <i>seconds</i>—Enters how often the switch sends keepalive messages. For <i>retry-count</i>, enters the number of unanswered keepalive messages that the switch sends before the connection is terminated. The default for each is 0. • (Optional) reconnect <i>time</i>—Enters the maximum time interval that the switch waits before trying to reconnect to the event gateway. • (Optional) source <i>ip-address</i>—Enters the source IP address of this device. <p>Note Though visible in the command-line help string, the encrypt and the clock-timeout <i>time</i> keywords are not supported.</p>
Step 3	end	Returns to privileged EXEC mode.
Step 4	show cns event connections	Verifies information about the event agent.

Enabling Cisco IOS CNS Agent and an Initial Configuration

	Command	Purpose
Step 1	configure terminal	Enters global configuration mode.
Step 2	cns template connect <i>name</i>	Enters CNS template connect configuration mode, and specifies the name of the CNS connect template.
Step 3	cli <i>config-text</i>	Enters a command line for the CNS connect template. Repeat this step for each command line in the template.
Step 4		Repeat Steps 2 to 3 to configure another CNS connect template.
Step 5	exit	Returns to global configuration mode.
Step 6	cns connect <i>name</i> [retries <i>number</i>] [retry-interval <i>seconds</i>] [sleep <i>seconds</i>] [timeout <i>seconds</i>]	Enters CNS connect configuration mode, specifies the name of the CNS connect profile, and defines the profile parameters. The switch uses the CNS connect profile to connect to Configuration Engine. <ul style="list-style-type: none"> (Optional) retries <i>number</i>—Enters the number of connection retries. The range is 1 to 30. The default is 3. (Optional) retry-interval <i>seconds</i>—Enters the interval between successive connection attempts to the Configuration Engine. The range is 1 to 40 seconds. The default is 10 seconds. (Optional) sleep <i>seconds</i>—Enters the amount of time before which the first connection attempt occurs. The range is 0 to 250 seconds. The default is 0. (Optional) timeout <i>seconds</i>—Enters the amount of time after which the connection attempts end. The range is 10 to 2000 seconds. The default is 120.
Step 7	discover { controller <i>controller-type</i> dcli [subinterface <i>subinterface-number</i>] interface [<i>interface-type</i>] line <i>line-type</i> }	Specifies the interface parameters in the CNS connect profile. <ul style="list-style-type: none"> controller <i>controller-type</i>—Enters the controller type. dcli—Enters the active data-link connection identifiers (DLCIs). (Optional) subinterface <i>subinterface-number</i>—Specifies the point-to-point subinterface number that is used to search for active DLCIs. interface [<i>interface-type</i>]—Enters the type of interface. line <i>line-type</i>—Enters the line type.
Step 8	template <i>name</i> [... <i>name</i>]	Specifies the list of CNS connect templates in the CNS connect profile to be applied to the switch configuration. You can specify more than one template.
Step 9		Repeat Steps 7 to 8 to specify more interface parameters and CNS connect templates in the CNS connect profile.
Step 10	exit	Returns to global configuration mode.
Step 11	hostname <i>name</i>	Enters the hostname for the switch.

	Command	Purpose
Step 12	ip route <i>network-number</i>	(Optional) Establishes a static route to Configuration Engine whose IP address is <i>network-number</i> .
Step 13	cns id <i>interface num</i> { dns-reverse ipaddress mac-address } [event] [image] or cns id { hardware-serial hostname string <i>string</i> udi } [event] [image]	(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. <ul style="list-style-type: none"> • <i>interface num</i>—Enters the type of interface for example, ethernet, group-async, loopback, or virtual-template. This setting specifies from which interface the IP or MAC address should be retrieved to define the unique ID. • dns-reverse—Retrieves the hostname and assigns it as the unique ID. • ipaddress—Uses the IP address. • mac-address—Uses the MAC address as the unique ID. • (Optional) event—Sets the ID to be the eventID value used to identify the switch. • (Optional) image—Sets the ID to be the imageID value used to identify the switch. <p>Note If the event and image keywords are omitted, the imageID value is used to identify the switch.</p> <ul style="list-style-type: none"> • hardware-serial—Sets the switch serial number as the unique ID. • hostname (the default)—Selects the switch hostname as the unique ID, uses an arbitrary text string string <i>string</i> as the unique ID and udi sets the unique device identifier (UDI) as the unique ID.

	Command	Purpose
Step 14	cns config initial { <i>hostname</i> <i>ip-address</i> } [<i>port-number</i>] [event] [no-persist] [page <i>page</i>] [source <i>ip-address</i>] [syntax-check]	<p>Enables the Cisco IOS agent and initiates an initial configuration.</p> <ul style="list-style-type: none"> • {<i>hostname</i> <i>ip-address</i>}—Enters the hostname or the IP address of the configuration server. • (Optional) <i>port-number</i>—Enters the port number of the configuration server. The default port number is 80. • (Optional) event—Enables configuration success, failure, or warning messages when the configuration is finished. • (Optional) no-persist—Suppresses the automatic writing to NVRAM of the configuration pulled as a result of entering the cns config initial global configuration command. If the no-persist keyword is not entered, using the cns config initial command causes the resultant configuration to be automatically written to NVRAM. • (Optional) page <i>page</i>—Enters the web page of the initial configuration. The default is /Config/config/asp. • (Optional) source <i>ip-address</i>—Enters the source IP address. • (Optional) syntax-check—Checks the syntax when this parameter is entered. <p> Note Though visible in the command-line help string, the encrypt, status url, and inventory keywords are not supported.</p>
Step 15	end	Returns to privileged EXEC mode.

Enabling a Partial Configuration

	Command	Purpose
Step 1	configure terminal	Enters global configuration mode.
Step 2	cns config partial { <i>ip-address</i> <i>hostname</i> } [<i>port-number</i>] [source <i>ip-address</i>]	<p>Enables the configuration agent, and initiates a partial configuration.</p> <ul style="list-style-type: none"> • {<i>ip-address</i> <i>hostname</i>}—Enters the IP address or the hostname of the configuration server. • (Optional) <i>port-number</i>—Enters the port number of the configuration server. The default port number is 80. • (Optional) source <i>ip-address</i>—Enters the source IP address. <p>Note Though visible in the command-line help string, the encrypt keyword is not supported.</p>
Step 3	end	Returns to privileged EXEC mode.

Monitoring and Maintaining Cisco IOS Configuration Engine

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

Configuration Examples for Cisco IOS Configuration Engine

Enabling the CNS Event Agent: Example

This example shows how to enable the CNS event agent, set the IP address gateway to 10.180.1.27, set 120 seconds as the keepalive interval, and set 10 as the retry count.

```
Switch(config)# cns event 10.180.1.27 keepalive 120 10
```

Configuring an Initial CNS Configuration: Examples

This example shows how to configure an initial configuration on a remote switch when the switch configuration is unknown (the CNS Zero Touch feature).

```
Switch(config)# cns template connect template-dhcp
Switch(config-tmpl-conn)# cli ip address dhcp
Switch(config-tmpl-conn)# exit
Switch(config)# cns template connect ip-route
Switch(config-tmpl-conn)# cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
Switch(config-tmpl-conn)# exit
Switch(config)# cns connect dhcp
Switch(config-cns-conn)# discover interface gigabitethernet
Switch(config-cns-conn)# template template-dhcp
Switch(config-cns-conn)# template ip-route
Switch(config-cns-conn)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# cns config initial 10.1.1.1 no-persist
```

This example shows how to configure an initial configuration on a remote switch when the switch IP address is known. The Configuration Engine IP address is 172.28.129.22.

```
Switch(config)# cns template connect template-dhcp
Switch(config-tmpl-conn)# cli ip address dhcp
Switch(config-tmpl-conn)# exit
Switch(config)# cns template connect ip-route
Switch(config-tmpl-conn)# cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
Switch(config-tmpl-conn)# exit
Switch(config)# cns connect dhcp
```

```
Switch(config-cns-conn)# discover interface gigabitethernet
Switch(config-cns-conn)# template template-dhcp
Switch(config-cns-conn)# template ip-route
Switch(config-cns-conn)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# ip route 172.28.129.22 255.255.255.255 11.11.11.1
RemoteSwitch(config)# cns id ethernet 0 ipaddress
RemoteSwitch(config)# cns config initial 172.28.129.22 no-persist
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IE 2000 or ESS-2020 commands	<i>Cisco IE 2000 Switch Command Reference</i> , Release 15.0(2)EC
Cisco IOS basic commands	<i>Cisco IOS Configuration Fundamentals Command Reference</i>
Network management commands	<i>Cisco IOS Network Management Command Reference</i>

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

