Configuring Keychain Management

This chapter describes how to configure keychain management on a Cisco NX-OS device.

This chapter includes the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your 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 "New and Changed Information"chapter or the Feature History table in this chapter.

Information About Keychain Management

Keychains and Keychain Management

Keychain management allows you to create and maintain keychains, which are sequences of keys (sometimes called shared secrets). You can use keychains with features that secure communications with other devices by using key-based authentication. The device allows you to configure multiple keychains.

Some routing protocols that support key-based authentication can use a keychain to implement a hitless key rollover for authentication. For more information, see the *Cisco Nexus 7000 Series NX-OS Unicast Routing Configuration Guide*.

Lifetime of a Key

To maintain stable communications, each device that uses a protocol that is secured by key-based authentication must be able to store and use more than one key for a feature at the same time. Based on the send and accept lifetimes of a key, keychain management provides a secure mechanism to handle key rollover. The device uses the lifetimes of keys to determine which keys in a keychain are active.

Each key in a keychain has two lifetimes, as follows:

Accept lifetime

The time interval within which the device accepts the key during a key exchange with another device.

Send lifetime

The time interval within which the device sends the key during a key exchange with another device.

You define the send and accept lifetimes of a key using the following parameters:

Start-time

The absolute time that the lifetime begins.

End-time

The end time can be defined in one of the following ways:

- The absolute time that the lifetime ends
- The number of seconds after the start time that the lifetime ends
- Infinite lifetime (no end-time)

During a key send lifetime, the device sends routing update packets with the key. The device does not accept communication from other devices when the key sent is not within the accept lifetime of the key on the device.

We recommend that you configure key lifetimes that overlap within every keychain. This practice avoids failure of neighbor authentication due to the absence of active keys.

Virtualization Support for Keychain Management

The following information applies to keychains used in virtual device contexts (VDCs):

- Keychains are unique per VDC. You cannot use a keychain that you created in one VDC in a different VDC.
- Because keychains are not shared by VDCs, you can reuse keychain names in different VDCs.
- The device does not limit keychains on a per-VDC basis.

Licensing Requirements for Keychain Management

This table shows the licensing requirements for keychain management.

Product	License Requirement
Cisco NX-OS	Keychain management requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For an explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Prerequisites for Keychain Management

Keychain management has no prerequisites.

Guidelines and Limitations for Keychain Management

Keychain management has the following configuration guideline and limitation:

- Changing the system clock impacts when the keys are active.

Default Settings for Keychain Management

This table lists the default settings for Cisco NX-OS keychain management parameters.

Table 1: Default Keychain Management Parameters

Parameters	Default
Key chains	No keychain exists by default.
Keys	No keys are created by default when you create a new keychain.
Accept lifetime	Always valid.
Send lifetime	Always valid.
Key-string entry encryption	Unencrypted.

Configuring Keychain Management

Creating a Keychain

You can create a keychain on the device. A new keychain contains no keys.

SUMMARY STEPS

1. **configure terminal**
2. **key chain** *name*
3. (Optional) **show key chain** *name*
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**key chain** *name*
	Example:
	`switch(config)# key chain glbp-keys`
	`switch(config-keychain)#`
		Creates the keychain and enters keychain configuration mode.

Step 3	(Optional) **show key chain** *name* **Example:** `switch(config-keychain)# show key chain glbp-keys`	Displays the keychain configuration.
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config-keychain)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring a Master Key and Enabling the AES Password Encryption Feature, on page 5

Removing a Keychain

You can remove a keychain on the device.

Note Removing a keychain removes any keys within the keychain.

Before you begin

If you are removing a keychain, ensure that no feature uses it. If a feature is configured to use a keychain that you remove, that feature is likely to fail to communicate with other devices.

SUMMARY STEPS

1. **configure terminal**

2. **no key chain** *name*
3. (Optional) **show key chain** *name*
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**no key chain** *name* **Example:** `switch(config)# no key chain glbp-keys`	Removes the keychain and any keys that the keychain contains.
Step 3	(Optional) **show key chain** *name* **Example:** `switch(config-keychain)# show key chain glbp-keys`	Confirms that the keychain no longer exists in running configuration.
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config-keychain)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Creating a Keychain, on page 3

Configuring a Master Key and Enabling the AES Password Encryption Feature

You can configure a master key for type-6 encryption and enable the Advanced Encryption Standard (AES) password encryption feature.

SUMMARY STEPS

1. [**no**] **key config-key ascii**
2. **configure terminal**
3. [**no**] **feature password encryption aes**
4. (Optional) **show encryption service stat**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	[**no**] **key config-key ascii** **Example:**	Configures a master key to be used with the AES password encryption feature. The master key can contain between 16

	Command or Action	Purpose

	`switch# key config-key ascii`
	`New Master Key:`
	`Retype Master Key:`
		and 32 alphanumeric characters. You can use the **no** form of this command to delete the master key at any time.
		If you enable the AES password encryption feature before configuring a master key, a message appears stating that password encryption will not take place unless a master key is configured. If a master key is already configured, you are prompted to enter the current master key before entering a new master key.

Step 2	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 3	[**no**] **feature password encryption aes** **Example:** `switch(config)# feature password encryption aes`	Enables or disables the AES password encryption feature.
Step 4	(Optional) **show encryption service stat** **Example:** `switch(config)# show encryption service stat`	Displays the configuration status of the AES password encryption feature and the master key.
Step 5	Required: **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration. **Note** This command is necessary to synchronize the master key in the running configuration and the startup configuration.

Related Topics

Configuring Text for a Key, on page 6
Configuring Accept and Send Lifetimes for a Key, on page 8
AES Password Encryption and Master Encryption Keys

Configuring Text for a Key

You can configure the text for a key. The text is the shared secret. The device stores the text in a secure format.

By default, accept and send lifetimes for a key are infinite, which means that the key is always valid. After you configure the text for a key, configure the accept and send lifetimes for the key.

Before you begin

Determine the text for the key. You can enter the text as unencrypted text or in the encrypted form that Cisco NX-OS uses to display key text when you use the **show key chain** command. Using the encrypted form is particularly helpful if you are creating key text to match a key as shown in the **show key chain** command output from another device.

SUMMARY STEPS

1. **configure terminal**
2. **key chain** *name*
3. **key** *key-ID*
4. **key-string** [*encryption-type*] *text-string*
5. (Optional) **show key chain** *name* [**mode decrypt**]
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**key chain** *name*
	Example:
	`switch(config)# key chain glbp-keys`
	`switch(config-keychain)#`
		Enters keychain configuration mode for the keychain that you specified.

Step 3	**key** *key-ID*
	Example:
	`switch(config-keychain)# key 13`
	`switch(config-keychain-key)#`
		Enters key configuration mode for the key that you specified. The *key-ID* argument must be a whole number between 0 and 65535.

Step 4	**key-string** [*encryption-type*] *text-string* **Example:** `switch(config-keychain-key)# key-string 0 AS3cureStr1ng`	Configures the text string for the key. The *text-string* argument is alphanumeric, case-sensitive, and supports special characters. The *encryption-type* argument can be one of the following values: • 0—The *text-string* argument that you enter is unencrypted text. This is the default. • 7—The *text-string* argument that you enter is encrypted. The encryption method is a Cisco proprietary method. This option is useful when you are entering a text string based on the encrypted output of a **show key chain** command that you ran on another Cisco NX-OS device. The value of the first 2 digits of a type 7 key string configured by using the **key-string 7** *text-string* command has to be between 0 and 15. For example, you can configure 07372b557e2c1a as the key string value in which case the sum value of the first 2 digits will be 7. But, you cannot configure 85782916342021 as the key string value because the value of the first 2 digits will be 85. We recommend

	Command or Action	Purpose
		unconfiguring any type 7 key strings that do not adhere to this value or to configure a type 0 string.
Step 5	(Optional) **show key chain** *name* [**mode decrypt**] **Example:** `switch(config-keychain-key)# show key chain glbp-keys`	Shows the keychain configuration, including the key text configuration. The mode decrypt option, which can be used by a device administrator only, displays the keys in cleartext.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config-keychain-key)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring Accept and Send Lifetimes for a Key, on page 8

Configuring Accept and Send Lifetimes for a Key

You can configure the accept lifetime and send lifetime for a key. By default, accept and send lifetimes for a key are infinite, which means that the key is always valid.

Note We recommend that you configure the keys in a keychain to have overlapping lifetimes. This practice prevents loss of key-secured communication due to moments where no key is active.

SUMMARY STEPS

1. **configure terminal**
2. **key chain** *name*
3. **key** *key-ID*
4. **accept-lifetime** [**local**] *start-time* **duration** *duration-value* | **infinite** | *end-time*]
5. **send-lifetime** [**local**] *start-time* **duration** *duration-value* | **infinite** | *end-time*]
6. (Optional) **show key chain** *name* [**mode decrypt**]
7. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

	Command or Action	Purpose

Step 2	**key chain** *name*
	Example:
	`switch(config)# key chain glbp-keys`
	`switch(config-keychain)#`
		Enters keychain configuration mode for the keychain that you specified.

Step 3	**key** *key-ID*
	Example:
	`switch(config-keychain)# key 13`
	`switch(config-keychain-key)#`
		Enters key configuration mode for the key that you specified.

Step 4	**accept-lifetime** [**local**] *start-time* **duration** *duration-value* | **infinite** | *end-time*] **Example:** `switch(config-keychain-key)# accept-lifetime 00:00:00 Jun 13 2008 23:59:59 Sep 12 2008`	Configures an accept lifetime for the key. By default, the device treats the *start-time* and *end-time* arguments as UTC. If you specify the **local** keyword, the device treats these times as local times. The *start-time* argument is the time of day and date that the key becomes active. Specify the end of the lifetime with one of the following options: • **duration** *duration-value* —The length of the lifetime in seconds. The maximum length is 2147483646 seconds (approximately 68 years). • **infinite**—The accept lifetime of the key never expires. • *end-time* —The *end-time* argument is the time of day and date that the key becomes inactive.
Step 5	**send-lifetime** [**local**] *start-time* **duration** *duration-value* | **infinite** | *end-time*] **Example:** `switch(config-keychain-key)# send-lifetime 00:00:00 Jun 13 2008 23:59:59 Aug 12 2008`	Configures a send lifetime for the key. By default, the device treats the *start-time* and *end-time* arguments as UTC. If you specify the **local** keyword, the device treats these times as local times. The *start-time* argument is the time of day and date that the key becomes active. You can specify the end of the send lifetime with one of the following options: • **duration** *duration-value* —The length of the lifetime in seconds. The maximum length is 2147483646 seconds (approximately 68 years). • **infinite**—The send lifetime of the key never expires. • *end-time* —The *end-time* argument is the time of day and date that the key becomes inactive.
Step 6	(Optional) **show key chain** *name* [**mode decrypt**] **Example:**	Shows the keychain configuration, including the key text configuration. The mode decrypt option, which can be used

	Command or Action	Purpose
	`switch(config-keychain-key)# show key chain glbp-keys`	by a device administrator only, displays the keys in cleartext.
Step 7	(Optional) **copy running-config startup-config** **Example:** `switch(config-keychain-key)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Lifetime of a Key, on page 2

Determining Active Key Lifetimes

To determine which keys within a keychain have active accept or send lifetimes, use the command in this table. For detailed information about the fields in the output from this command, see the *Cisco Nexus 7000 Series NX-OS Security Command Reference*.

Command	Purpose
show key chain	Displays the keychains configured on the device.

Verifying the Keychain Management Configuration

To display keychain management configuration information, perform the following task. For detailed information about the fields in the output from this command, see the *Cisco Nexus 7000 Series NX-OS Security Command Reference*.

Command	Purpose
show key chain	Displays the keychains configured on the device.

Configuration Example for Keychain Management

This example shows how to configure a keychain named glbp keys. Each key text string is encrypted. Each key has longer accept lifetimes than send lifetimes, to help prevent lost communications by accidentally configuring a time in which there are no active keys.

```
key chain glbp-keys
  key 0
    key-string 7 zqdest
    accept-lifetime 00:00:00 Jun 01 2008 23:59:59 Sep 12 2008
    send-lifetime 00:00:00 Jun 01 2008 23:59:59 Aug 12 2008
  key 1
    key-string 7 uaeqdyito
    accept-lifetime 00:00:00 Aug 12 2008 23:59:59 Dec 12 2008
    send-lifetime 00:00:00 Sep 12 2008 23:59:59 Nov 12 2008
```

```
key 2
  key-string 7 eekgsdyd
  accept-lifetime 00:00:00 Nov 12 2008 23:59:59 Mar 12 2009
  send-lifetime 00:00:00 Dec 12 2008 23:59:59 Feb 12 2009
```

Where to Go Next

For information about routing features that use keychains, see the *Cisco Nexus 7000 Series NX-OS Unicast Routing Configuration Guide*.

Additional References for Keychain Management

Related Documents

Related Topic	Document Title
Gateway Load Balancing Protocol	*Cisco Nexus 7000 Series NX-OS Unicast Routing Configuration Guide*
Border Gateway Protocol	*Cisco Nexus 7000 Series NX-OS Unicast Routing Configuration Guide*
Keychain management commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 7000 Series NX-OS Security Command Reference*

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.	—