Cisco Nexus 3550-T NX-OS Security Configuration Guide, Release 10.2(x)

First Published: 2022-09-19

Last Modified: 2023-05-26

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

Any products and features described herein as in development or available at a future date remain in varying stages of development and will be offered on a when-and if-available basis. Any such product or feature roadmaps are subject to change at the sole discretion of Cisco and Cisco will have no liability for delay in the delivery or failure to deliver any products or feature roadmap items that may be set forth in this document.

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.

The documentation set for this product strives to use bias-free language. For the purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

CONTENTS

Trademarks ?

PREFACE Preface ix

- Audience ix
- Document Conventions ix
- Related Documentation for Cisco Nexus 3550-T Switches x
- Documentation Feedback x
- Communications, Services, and Additional Information x

CHAPTER 1 New and Changed Information 1

- New and Changed Information 1

CHAPTER 2 Cisco Nexus 3550-T Security Configuration Overview 3

- Authentication, Authorization, and Accounting 3
- RADIUS and TACACS+ Security Protocols 4
- SSH and Telnet 4
- IP ACLs 5
- Rate Limiters 5

CHAPTER 3 Configuring AAA 7

- About AAA 7
  - AAA Security Services 7
  - Benefits of Using AAA 8
  - Remote AAA Services 8
  - AAA Server Groups 9
  - AAA Service Configuration Options 9

Authentication and Authorization Process for User Login 10

AES Password Encryption and Primary Encryption Keys 11

Prerequisites for AAA 11

Guidelines and Limitations for AAA 11

Default Settings for AAA 12

Configuring AAA 12

Process for Configuring AAA 12

Configuring Console Login Authentication Methods 13

Configuring Default Login Authentication Methods 14

Disabling Fallback to Local Authentication 16

Enabling the Default User Role for AAA Authentication 17

Enabling Login Authentication Failure Messages 18

Logging Successful and Failed Login Attempts 19

Configuring Login Block Per User 20

Enabling CHAP Authentication 22

Enabling MSCHAP or MSCHAP V2 Authentication 23

Configuring AAA Accounting Default Methods 25

Using AAA Server VSAs with Cisco NX-OS Devices 26

About VSAs 27

VSA Format 27

Specifying Cisco NX-OS User Roles and SNMPv3 Parameters on AAA Servers 28

Configuring Secure Login Features 28

Configuring Login Parameters 28

Restricting User Login Sessions 29

Restricting the Password Length 30

Enabling the Password Prompt for the Username 31

Configuring the Shared Secret for RADIUS or TACACS+ 31

Monitoring and Clearing the Local AAA Accounting Log 32

Verifying the AAA Configuration 32

Configuration Examples for AAA 33

Configuration Examples for Login Parameters 34

Configuration Examples for the Password Prompt Feature 34

Additional References for AAA 35

CHAPTER 4

Configuring RADIUS 37
About RADIUS 37
RADIUS Network Environments 37
RADIUS Operation 38
RADIUS Server Monitoring 38
Vendor-Specific Attributes 39
About RADIUS Change of Authorization 40
Session Reauthentication 41
Session Termination 41
Prerequisites for RADIUS 41
Guidelines and Limitations for RADIUS 41
Default Settings for RADIUS 42
Configuring RADIUS Servers 42
RADIUS Server Configuration Process 43
Configuring RADIUS Server Hosts 43
Configuring Global RADIUS Keys 44
Configuring a Key for a Specific RADIUS Server 45
Configuring RADIUS Server Groups 47
Configuring the Global Source Interface for RADIUS Server Groups 48
Allowing Users to Specify a RADIUS Server at Login 49
Configuring the Global RADIUS Transmission Retry Count and Timeout Interval 50
Configuring the RADIUS Transmission Retry Count and Timeout Interval for a Server 51
Configuring Accounting and Authentication Attributes for RADIUS Servers 53
Configuring Global Periodic RADIUS Server Monitoring 54
Configuring Periodic RADIUS Server Monitoring on Individual Servers 56
Configuring the RADIUS Dead-Time Interval 57
Configuring One-Time Passwords 58
Manually Monitoring RADIUS Servers or Groups 59
Enabling or Disabling Dynamic Author Server 59
Configuring RADIUS Change of Authorization 60
Verifying the RADIUS Configuration 60
Verifying RADIUS Change of Authorization Configuration 61
Monitoring RADIUS Servers 61

Clearing RADIUS Server Statistics 62
Configuration Example for RADIUS 62
Configuration Examples of RADIUS Change of Authorization 63
Additional References for RADIUS 63

CHAPTER 5 **Configuring IP ACLs** 65
About ACLs 65
ACL Types and Applications 65
Order of ACL Application 66
About Rules 66
Protocols for IP ACLs 67
Source and Destination 67
Implicit Rules for IP ACL 67
Additional Filtering Options 67
Sequence Numbers 67
Logical Operators and Logical Operation Units 68
Session Manager Support for IP ACLs 68
Prerequisites for IP ACLs 68
Guidelines and Limitations for IP ACLs 69
Default Settings for IP ACLs 70
Configuring IP ACLs 70
Creating an IP ACL 70
Changing an IP ACL 71
Changing Sequence Numbers in an IP ACL 72
Removing an IP ACL 73
Applying an IP ACL as a Router ACL 74
Verifying the IP ACL Configuration 75
Configuration Examples for IP ACLs 76
Verifying the Object-Group Configuration 77
Verifying the Time-Range Configuration 77

CHAPTER 6 **Configuring SSH and Telnet** 79
About SSH and Telnet 79
SSH Server 79

SSH Client 79

SSH Server Keys 80

SSH Authentication Using Digital Certificates 80

Telnet Server 81

Prerequisites for SSH and Telnet 81

Guidelines and Limitations for SSH and Telnet 81

Default Settings for SSH and Telnet 82

Configuring SSH 82

Generating SSH Server Keys 82

Specifying the SSH Public Keys for User Accounts 83

Specifying the SSH Public Keys in IETF SECSH Format 83

Specifying the SSH Public Keys in OpenSSH Format 84

Configuring a Maximum Number of SSH Login Attempts 85

Starting SSH Sessions 86

Starting SSH Sessions from Boot Mode 86

Configuring SSH Passwordless File Copy 87

Configuring SCP and SFTP Servers 89

Configuring X.509v3 Certificate-Based SSH Authentication 90

Configuring Legacy SSH Algorithm Support 92

Algorithms Supported - FIPs Mode Enabled 93

Changing the Default SSH Server Port 94

Clearing SSH Hosts 95

Disabling the SSH Server 95

Deleting SSH Server Keys 96

Clearing SSH Sessions 97

Configuring Telnet 97

Enabling the Telnet Server 97

Starting Telnet Sessions to Remote Devices 98

Clearing Telnet Sessions 98

Verifying the SSH and Telnet Configuration 99

Configuration Example for SSH 99

Configuration Example for SSH Passwordless File Copy 100

Configuration Example for X.509v3 Certificate-Based SSH Authentication 102

Additional References for SSH and Telnet 103

CHAPTER 7 Configuring DHCP 105

About DHCP Client 105

Guidelines and Limitations for DHCP 105

Enabling DHCP Client 105

Configuration Examples for DHCP Client 106

Preface

This preface includes the following sections:

- Audience, on page ix
- Document Conventions, on page ix
- Related Documentation for Cisco Nexus 3550-T Switches, on page x
- Documentation Feedback, on page x
- Communications, Services, and Additional Information, on page x

Audience

This publication is for network administrators who install, configure, and maintain Cisco Nexus switches.

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
Italic	Italic text indicates arguments for which you supply the values.
[x]	Square brackets enclose an optional element (keyword or argument).
[x | y]	Square brackets enclosing keywords or arguments that are separated by a vertical bar indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments that are separated by a vertical bar indicate a required choice.
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.

Convention	Description
`variable`	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string includes the quotation marks.

Examples use the following conventions:

Convention	Description
`screen font`	Terminal sessions and information the switch displays are in screen font.
`boldface screen font`	Information that you must enter is in boldface screen font.
italic screen font	Arguments for which you supply values are in italic screen font.
< >	Nonprinting characters, such as passwords, are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Related Documentation for Cisco Nexus 3550-T Switches

The entire Cisco Nexus 3550-T switch documentation set is available at the following URL:

https://www.cisco.com/c/en/us/support/switches/nexus-3550-series/series.html

Documentation Feedback

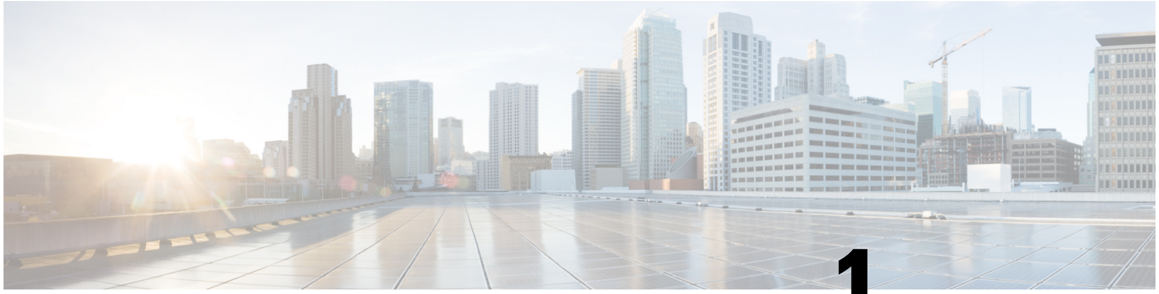
To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus9k-docfeedback@cisco.com. We appreciate your feedback.

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.



CHAPTER 1

New and Changed Information

This section contains the new and changed information.

- New and Changed Information, on page 1

New and Changed Information

Table 1: New and Changed Information for Cisco Nexus 3550-T NX-OS Release 10.2(x)

Feature	Description	Changed in Release	Where Documented
Traffic rate limiters	Hardware rate-limiters protect the supervisor CPU from excessive inbound traffic. Rate limits are embedded on each port of an NX-OS device.	10.2(3t)	Rate Limiters, on page 5
ACL support on switch virtual interfaces (SVI)	ACL support extended to SVIs.	10.2(3t)	ACL Types and Applications, on page 65

CHAPTER 2

Cisco Nexus 3550-T Security Configuration Overview

The Cisco NX-OS software supports security features that can protect your network against degradation or failure and also against data loss or compromise resulting from intentional attacks and from unintended but damaging mistakes by well-meaning network users.

This chapter includes the following sections:

- Authentication, Authorization, and Accounting, on page 3
- RADIUS and TACACS+ Security Protocols, on page 4
- SSH and Telnet, on page 4
- IP ACLs, on page 5
- Rate Limiters, on page 5

Authentication, Authorization, and Accounting

Authentication, authorization, and accounting (AAA) is an architectural framework for configuring a set of three independent security functions in a consistent, modular manner.

Authentication

Provides the method of identifying users, including login and password dialog, challenge and response, messaging support, and, depending on the security protocol that you select, encryption. Authentication is the way a user is identified prior to being allowed access to the network and network services. You configure AAA authentication by defining a named list of authentication methods and then applying that list to various interfaces.

Authorization

Provides the method for remote access control, including one-time authorization or authorization for each service, per-user account list and profile, user group support, and support of IP, IPX, ARA, and Telnet.

Remote security servers, such as RADIUS and TACACS+, authorize users for specific rights by associating attribute-value (AV) pairs, which define those rights, with the appropriate user. AAA authorization works by assembling a set of attributes that describe what the user is authorized to perform. These attributes are compared with the information contained in a database for a given user, and the result is returned to AAA to determine the user’s actual capabilities and restrictions.

Accounting
Provides the method for collecting and sending security server information used for billing, auditing, and reporting, such as user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes. Accounting enables you to track the services that users are accessing, as well as the amount of network resources that they are consuming.

Note You can configure authentication outside of AAA. However, you must configure AAA if you want to use RADIUS or TACACS+, or if you want to configure a backup authentication method.

For more information, see the Configuring AAA, on page 7 chapter.

RADIUS and TACACS+ Security Protocols

AAA uses security protocols to administer its security functions. If your router or access server is acting as a network access server, AAA is the means through which you establish communication between your network access server and your RADIUS security server.

The chapters in this guide describe how to configure the following security server protocols:

RADIUS
A distributed client/server system implemented through AAA that secures networks against unauthorized access. In the Cisco implementation, RADIUS clients run on Cisco routers and send authentication requests to a central RADIUS server that contains all user authentication and network service access information.

TACACS+
A security application implemented through AAA that provides a centralized validation of users who are attempting to gain access to a router or network access server. TACACS+ services are maintained in a database on a TACACS+ daemon running, typically, on a UNIX or Windows NT workstation. TACACS+ provides for separate and modular authentication, authorization, and accounting facilities.

For more information, see the Configuring RADIUS, on page 37 chapter.

SSH and Telnet

You can use the Secure Shell (SSH) server to enable an SSH client to make a secure, encrypted connection to a Cisco NX-OS device. SSH uses strong encryption for authentication. The SSH server in the Cisco NX-OS software can interoperate with publicly and commercially available SSH clients.

The SSH client in the Cisco NX-OS software works with publicly and commercially available SSH servers.

The Telnet protocol enables TCP/IP connections to a host. Telnet allows a user at one site to establish a TCP connection to a login server at another site and then passes the keystrokes from one device to the other. Telnet can accept either an IP address or a domain name as the remote device address.

For more information, see the Configuring SSH and Telnet, on page 79 chapter.

IP ACLs

IP ACLs are ordered sets of rules that you can use to filter traffic based on IPv4 information in the Layer 3 header of packets. Each rule specifies a set of conditions that a packet must satisfy to match the rule. When the Cisco NX-OS software determines that an IP ACL applies to a packet, it tests the packet against the conditions of all rules. The first match determines whether a packet is permitted or denied, or if there is no match, the Cisco NX-OS software applies the applicable default rule. The Cisco NX-OS software continues processing packets that are permitted and drops packets that are denied.

For more information, see the Configuring IP ACLs, on page 65 chapter.

Rate Limiters

Hardware rate-limiters protect the supervisor CPU from excessive inbound traffic. Rate limiters are embedded on each port of an NX-OS device. The same rate limiter value is applied on every port of the device, and this value cannot be changed or configured.

Note The storm control command, **storm-control-cpu all** *rate*, supported in the Cisco 3550-T NX-OS release 10.1(2t), is not supported on release 10.2(3t), as the CPU traffic is controlled by the rate limiters in the 10.2(3t) release.

CHAPTER 3

Configuring AAA

This chapter describes how to configure authentication, authorization, and accounting (AAA) on Cisco NX-OS devices.

This chapter includes the following sections:

- About AAA, on page 7
- Prerequisites for AAA, on page 11
- Guidelines and Limitations for AAA, on page 11
- Default Settings for AAA, on page 12
- Configuring AAA, on page 12
- Monitoring and Clearing the Local AAA Accounting Log , on page 32
- Verifying the AAA Configuration, on page 32
- Configuration Examples for AAA, on page 33
- Configuration Examples for Login Parameters, on page 34
- Configuration Examples for the Password Prompt Feature, on page 34
- Additional References for AAA, on page 35

About AAA

This section includes information about AAA on Cisco NX-OS devices.

AAA Security Services

The AAA feature allows you to verify the identity of, grant access to, and track the actions of users managing a Cisco NX-OS device. Cisco NX-OS devices support Remote Access Dial-In User Service (RADIUS) or Terminal Access Controller Access Control System Plus (TACACS+) protocols.

Based on the user ID and password combination that you provide, Cisco NX-OS devices perform local authentication or authorization using the local database or remote authentication or authorization using one or more AAA servers. A preshared secret key provides security for communication between the Cisco NX-OS device and AAA servers. You can configure a common secret key for all AAA servers or for only a specific AAA server.

AAA security provides the following services:

Authentication

Identifies users, including login and password dialog, challenge and response, messaging support, and, depending on the security protocol that you select, encryption.

Authentication is the process of verifying the identity of the person or device accessing the Cisco NX-OS device, which is based on the user ID and password combination provided by the entity trying to access the Cisco NX-OS device. Cisco NX-OS devices allow you to perform local authentication (using the local lookup database) or remote authentication (using one or more RADIUS or TACACS+ servers).

Authorization

Provides access control.AAA authorization is the process of assembling a set of attributes that describe what the user is authorized to perform. Authorization in the Cisco NX-OS software is provided by attributes that are downloaded from AAA servers. Remote security servers, such as RADIUS and TACACS+, authorize users for specific rights by associating attribute-value (AV) pairs, which define those rights with the appropriate user.

Accounting

Provides the method for collecting information, logging the information locally, and sending the information to the AAA server for billing, auditing, and reporting.

The accounting feature tracks and maintains a log of every management session used to access the Cisco NX-OS device. You can use this information to generate reports for troubleshooting and auditing purposes. You can store accounting logs locally or send them to remote AAA servers.

Note The Cisco NX-OS software supports authentication, authorization, and accounting independently. For example, you can configure authentication and authorization without configuring accounting.

Benefits of Using AAA

AAA provides the following benefits:

- Increased flexibility and control of access configuration
- Scalability
- Standardized authentication methods, such as RADIUS and TACACS+
- Multiple backup devices

Remote AAA Services

Remote AAA services provided through RADIUS and TACACS+ protocols have the following advantages over local AAA services:

- It is easier to manage user password lists for each Cisco NX-OS device in the fabric.
- AAA servers are already deployed widely across enterprises and can be easily used for AAA services.
- You can centrally manage the accounting log for all Cisco NX-OS devices in the fabric.
- It is easier to manage user attributes for each Cisco NX-OS device in the fabric than using the local databases on the devices.

AAA Server Groups

You can specify remote AAA servers for authentication, authorization, and accounting using server groups. A server group is a set of remote AAA servers that implements the same AAA protocol. The purpose of a server group is to provide for failover servers in case a remote AAA server fails to respond. If the first remote server in the group fails to respond, the next remote server in the group is tried until one of the servers sends a response. If all the AAA servers in the server group fail to respond, then that server group option is considered a failure. If required, you can specify multiple server groups. If the Cisco NX-OS device encounters errors from the servers in the first group, it tries the servers in the next server group.

AAA Service Configuration Options

The AAA configuration in Cisco NX-OS devices is service based, which means that you can have separate AAA configurations for the following services:

- User Telnet or Secure Shell (SSH) login authentication
- Console login authentication
- Extensible Authentication Protocol over User Datagram Protocol (EAPoUDP) authentication for Network Admission Control (NAC)
- User management session accounting

This table provides the related CLI command for each AAA service configuration option.

Table 2: AAA Service Configuration Commands

AAA Service Configuration Option	Related Command
Telnet or SSH login	**aaa authentication login default**
Console login	**aaa authentication login console**
	aaa authentication eou default
User session accounting	**aaa accounting default**

You can specify the following authentication methods for the AAA services:

All RADIUS servers

Uses the global pool of RADIUS servers for authentication.

Specified server groups

Uses specified RADIUS, TACACS+, or LDAP server groups you have configured for authentication.

Local

Uses the local username or password database for authentication.

None

Specifies that no AAA authentication be used.

Note If you specify the all RADIUS servers method, rather than a specified server group method, the Cisco NX-OS device chooses the RADIUS server from the global pool of configured RADIUS servers, in the order of configuration. Servers from this global pool are the servers that can be selectively configured in a RADIUS server group on the Cisco NX-OS device.

This table shows the AAA authentication methods that you can configure for the AAA services.

Table 3: AAA Authentication Methods for AAA Services

AAA Service	AAA Methods
Console login authentication	Server groups, local, and none
User login authentication	Server groups, local, and none
User management session accounting	Server groups and local

Note For console login authentication, user login authentication, and user management session accounting, the Cisco NX-OS device tries each option in the order specified. The local option is the default method when other configured options fail. You can disable the local option for the console or default login by using the **no aaa authentication login** {**console** | **default**} **fallback error local** command.

Authentication and Authorization Process for User Login

The following list explains the process:

- When you log in to the required Cisco NX-OS device, you can use the Telnet, SSH, or console login options.
- When you have configured the AAA server groups using the server group authentication method, the Cisco NX-OS device sends an authentication request to the first AAA server in the group as follows:
  - If the AAA server fails to respond, the next AAA server is tried and so on until the remote server responds to the authentication request.
  - If all AAA servers in the server group fail to respond, the servers in the next server group are tried.
  - If all configured methods fail, the local database is used for authentication, unless fallback to local is disabled for the console login.
- If the Cisco NX-OS device successfully authenticates you through a remote AAA server, then the following possibilities apply:
  - If the AAA server protocol is RADIUS, then user roles specified in the cisco-av-pair attribute are downloaded with an authentication response.
  - If the AAA server protocol is TACACS+, then another request is sent to the same server to get the user roles specified as custom attributes for the shell.

- If your username and password are successfully authenticated locally, the Cisco NX-OS device logs you in and assigns you the roles configured in the local database.

Note "No more server groups left" means that there is no response from any server in all server groups. "No more servers left" means that there is no response from any server within this server group.

AES Password Encryption and Primary Encryption Keys

You can enable strong, reversible 128-bit Advanced Encryption Standard (AES) password encryption, also known as type-6 encryption. To start using type-6 encryption, you must enable the AES password encryption feature and configure a primary encryption key, which is used to encrypt and decrypt passwords.

After you enable AES password encryption and configure a primary key, all existing and newly created clear-text passwords for supported applications (currently RADIUS and TACACS+) are stored in type-6 encrypted format, unless you disable type-6 password encryption. You can also configure Cisco NX-OS to convert all existing weakly encrypted passwords to type-6 encrypted passwords.

Prerequisites for AAA

Remote AAA servers have the following prerequisites:

- Ensure that at least one RADIUS, TACACS+, or LDAP server is reachable through IP.
- Ensure that the Cisco NX-OS device is configured as a client of the AAA servers.
- Ensure that the secret key is configured on the Cisco NX-OS device and the remote AAA servers.
- Ensure that the remote server responds to AAA requests from the Cisco NX-OS device.

Guidelines and Limitations for AAA

AAA has the following guidelines and limitations:

- If you have a user account that is configured on the local Cisco NX-OS device that has the same name as a remote user account on an AAA server, the Cisco NX-OS software applies the user roles for the local user account to the remote user, not the user roles configured on the AAA server.
- Cisco Nexus® 3550-T switches support the **aaa authentication login ascii-authentication** command only for TACACS+ (and not for RADIUS).
- If you modify the default login authentication method (without using the **local** keyword), the configuration overrides the console login authentication method. To explicitly configure the console authentication method, use the **aaa authentication login console** {**group** *group-list* [**none**] | **local** | **none**} command.
- The **login block-for** and **login quiet-mode** configuration mode commands are renamed to **system login block-for** and **system login quiet-mode**, respectively.

- When you use the **system login quiet-mode access-class QUIET_LIST** command, you must ensure that the access list is correctly defined to only block the specified traffic. For example, if you need to block only the user logins from untrusted hosts, then the access list should specify ports 22, 23, 80, and 443 corresponding to SSH, telnet, and HTTP-based access from those hosts.

Default Settings for AAA

This table lists the default settings for AAA parameters.

Table 4: Default AAA Parameter Settings

Parameters	Default
Console authentication method	local
Default authentication method	local
Login authentication failure messages	Disabled
CHAP authentication	Disabled
MSCHAP authentication	Disabled
Default accounting method	local
Accounting log display length	250 KB

Configuring AAA

This section describes the tasks for configuring AAA on Cisco NX-OS devices.

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Note Cisco Nexus® 3550-T Series switches support the CLI command, aaa authentication login ascii-authentication, only for TACAAS+, but not for RADIUS. Ensure that you have disabled aaa authentication login ascii-authentication switch so that the default authentication, PAP, is enabled. Otherwise, you will see syslog errors.

Process for Configuring AAA

Follow these steps to configure AAA authentication and accounting:

1. If you want to use remote RADIUS, TACACS+, or LDAP servers for authentication, configure the hosts on your Cisco NX-OS device.
2. Configure console login authentication methods.
3. Configure default login authentication methods for user logins.
4. Configure default AAA accounting default methods.

Configuring Console Login Authentication Methods

This section describes how to configure the authentication methods for the console login.

The authentication methods include the following:

- Global pool of RADIUS servers
- Named subset of RADIUS, TACACS+, or LDAP servers
- Local database on the Cisco NX-OS device
- Username only (none)

The default method is local, but you have the option to disable it.

Note The **group radius** and **group** *server-name* forms of the **aaa authentication** command refer to a set of previously defined RADIUS servers. Use the **radius-server host** command to configure the host servers. Use the **aaa group server radius** command to create a named group of servers.

Note If you perform a password recovery when remote authentication is enabled, local authentication becomes enabled for console login as soon as the password recovery is done. As a result, you can log into the Cisco NX-OS device through the console port using the new password. After login, you can continue to use local authentication, or you can enable remote authentication after resetting the admin password configured at the AAA servers. For more information about the password recovery process, see the *Cisco Nexus® Series NX-OS Troubleshooting Guide.*

Before you begin

Configure RADIUS, TACACS+, or LDAP server groups, as needed.

Procedure

	Command or Action	Purpose

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

	Command or Action	Purpose
Step 2	**aaa authentication login console** {**group** *group-list* [**none**] | **local** | **none**} **Example:** `switch(config)# aaa authentication login console group radius`	Configures login authentication methods for the console. The *group-list* argument consists of a space-delimited list of group names. The group names are the following: **radius** Uses the global pool of RADIUS servers for authentication. ***named-group*** Uses a named subset of RADIUS, TACACS+, or LDAP servers for authentication. The **local** method uses the local database for authentication, and the **none** method specifies that no AAA authentication be used. The default console login method is **local**, which is used when no methods are configured or when all the configured methods fail to respond, unless fallback to local is disabled for the console login.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 4	(Optional) **show aaa authentication** **Example:** `switch# show aaa authentication`	Displays the configuration of the console login authentication methods.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring Default Login Authentication Methods

The authentication methods include the following:

- Global pool of RADIUS servers
- Named subset of RADIUS, TACACS+, or LDAP servers
- Local database on the Cisco NX-OS device
- Username only

The default method is local, but you have the option to disable it.

Before you begin

Configure RADIUS, TACACS+, or LDAP server groups, as needed.

Procedure

	Command or Action	Purpose

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

Step 2	**aaa authentication login default** {**group** *group-list* [**none**] | **local** | **none**} **Example:** `switch(config)# aaa authentication login default group radius`	Configures the default authentication methods. The *group-list* argument consists of a space-delimited list of group names. The group names are the following: • **radius**—Uses the global pool of RADIUS servers for authentication. • *named-group*—Uses a named subset of RADIUS, TACACS+, or LDAP servers for authentication. The **local** method uses the local database for authentication, and the **none** method specifies that no AAA authentication be used. The default login method is **local**, which is used when no methods are configured or when all the configured methods fail to respond, unless fallback to local is disabled for the console login. You can configure one of the following: • AAA authentication groups • AAA authentication groups with no authentication • Local authentication • No authentication

	Command or Action	Purpose
		Note The **local** keyword is not supported (and is not required) when configuring AAA authentication groups because local authentication is the default if remote servers are unreachable. For example, if you configure **aaa authentication login default group g1**, local authentication is tried if you are unable to authenticate using AAA group g1. In contrast, if you configure **aaa authentication login default group g1 none**, no authentication is performed if you are unable to authenticate using AAA group g1.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 4	(Optional) **show aaa authentication** **Example:** `switch# show aaa authentication`	Displays the configuration of the default login authentication methods.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Disabling Fallback to Local Authentication

By default, if remote authentication is configured for console or default login and all AAA servers are unreachable (resulting in an authentication error), the Cisco NX-OS device falls back to local authentication to ensure that users aren’t locked out of the device. However, you can disable fallback to local authentication in order to increase security.

Caution Disabling fallback to local authentication can lock your Cisco NX-OS device, forcing you to perform a password recovery in order to gain access. To prevent being locked out of the device, we recommend that you disable fallback to local authentication for only the default login or the console login, not both.

Before you begin

Configure remote authentication for the console or default login.

Procedure

	Command or Action	Purpose

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

Step 2	**no aaa authentication login** {**console** | **default**} **fallback error local** **Example:** `switch(config)# no aaa authentication login console fallback error local`	Disables fallback to local authentication for the console or default login if remote authentication is configured and all AAA servers are unreachable. The following message appears when you disable fallback to local authentication: `“WARNING!!! Disabling fallback can lock your switch.”`

Step 3	(Optional) **exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 4	(Optional) **show aaa authentication** **Example:** `switch# show aaa authentication`	Displays the configuration of the console and default login authentication methods.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Enabling the Default User Role for AAA Authentication

You can allow remote users who do not have a user role to log in to the Cisco NX-OS device through a RADIUS or TACACS+ remote authentication server using a default user role. When you disable the AAA default user role feature, remote users who do not have a user role cannot log in to the device.

Procedure

	Command or Action	Purpose

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

Step 2	**aaa user default-role** **Example:** `switch(config)# aaa user default-role`	Enables the default user role for AAA authentication. The default is enabled. You can disable the default user role feature by using the **no** form of this command.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 4	(Optional) **show aaa user default-role** **Example:** `switch# show aaa user default-role`	Displays the AAA default user role configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Enabling Login Authentication Failure Messages

When you log in, the login is processed by rolling over to the local user database if the remote AAA servers do not respond. In such cases, the following messages display on the user’s terminal if you have enabled login failure messages:

```
Remote AAA servers unreachable; local authentication done.
```

```
Remote AAA servers unreachable; local authentication failed.
```

Procedure

	Command or Action	Purpose

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

Step 2	**aaa authentication login error-enable** **Example:**	Enables login authentication failure messages. The default is disabled.

	Command or Action	Purpose
	`switch(config)# aaa authentication login error-enable`	

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 4	(Optional) **show aaa authentication** **Example:** `switch# show aaa authentication`	Displays the login failure message configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Logging Successful and Failed Login Attempts

You can configure the switch to log all successful and failed login attempts to the configured syslog server.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	Required: [**no**] **login on-failure log** **Example:** `switch(config)# login on-failure log`	Logs all failed authentication messages to the configured syslog server only if the logging level is set to 6. With this configuration, the following syslog message appears after the failed login: AUTHPRIV-3-SYSTEM_MSG: pam_aaa:Authentication failed for user admin from 172.22.00.00 **Note** When logging level authpriv is 6, additional Linux kernel authentication messages appear along with the previous message. If these additional messages need to be ignored, the authpriv value should be set to 3.

	Command or Action	Purpose

Step 3	Required: [**no**] **login on-success log**
	Example:
	`switch(config)# login on-success log`
	`switch(config)# logging level authpriv 6`
	`switch(config)# logging level daemon 6`
		Logs all successful authentication messages to the configured syslog server only if the logging level is set to 6. With this configuration, the following syslog message appears after the successful login:
		AUTHPRIV-6-SYSTEM_MSG: pam_aaa:Authentication success for user admin from 172.22.00.00
		Note When logging level authpriv is 6, additional Linux kernel authentication messages appear along with the previous message. If these additional messages need to be ignored, the authpriv value should be set to 3.

Step 4	(Optional) **show login on-failure log** **Example:** `switch(config)# show login on-failure log`	Displays whether the switch is configured to log failed authentication messages to the syslog server.
Step 5	(Optional) **show login on-successful log** **Example:** `switch(config)# show login on-successful log`	Displays whether the switch is configured to log successful authentication messages to the syslog server.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring Login Block Per User

Ensure that the switch is in global configuration mode.

The Login Block Per User feature helps detect suspected Denial of Service (DoS) attacks and to slow down dictionary attacks. This feature is applicable for local users and remote users. Use this task to configure login parameters to block a user after failed login attempts.

Note You can configure login block for remote users.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**aaa authentication rejected** *attempts***in***seconds***ban***seconds* **Example:** `switch(config)# aaa authentication rejected 3 in 20 ban 300`	Configures login parameters to block a user. **Note** Use **no aaa authentication rejected** command to revert to the default login parameters.
Step 3	**exit** **Example:** `switch(config)# exit`	Exits to privileged EXEC mode.
Step 4	(Optional) **show running config** **Example:** `switch# show running config`	Displays the login parameters.
Step 5	**show aaa local user blocked** **Example:** `switch# show aaa local user blocked`	Displays the blocked local users.
Step 6	**clear aaa local user blocked {username user| all}** **Example:** `switch(config)# switch# clear aaa local user blocked username testuser`	Clears the blocked local users. all –Clears all the blocked local users.
Step 7	**show aaa user blocked** **Example:** `switch(config)# show aaa user blocked`	Displays all blocked local and remote users.
Step 8	(Optional) **clear aaa user blocked{username user| all}** **Example:** `switch# clear aaa user blocked username testuser`	Clears all blocked local and remote users. all – Clears all the blocked local and remote users.

Example

Note Only network-admin have privileges to run the show and clear commands.

The following example shows how to configure the login parameters to block a user for 300 seconds when three login attempts fail within a period of 20 seconds:

```
switch(config)# aaa authentication rejected 3 in 20 ban 300
switch# show run | i rejected
aaa authentication rejected 3 in 20 ban 300
switch# show aaa local user blocked
Local-user                 State
testuser                   Watched (till 11:34:42 IST Nov 12 2020)
switch# clear aaa local user blocked username testuser
switch# show aaa user blocked
Local-user                 State
testuser                   Watched (till 11:34:42 IST Nov 12 2020)
switch# clear aaa user blocked username testuser
```

Enabling CHAP Authentication

The Cisco NX-OS software supports the Challenge Handshake Authentication Protocol (CHAP), a challenge-response authentication protocol that uses the industry-standard Message Digest (MD5) hashing scheme to encrypt responses. You can use CHAP for user logins to a Cisco NX-OS device through a remote authentication server (RADIUS or TACACS+).

By default, the Cisco NX-OS device uses Password Authentication Protocol (PAP) authentication between the Cisco NX-OS device and the remote server. If you enable CHAP, you need to configure your RADIUS or TACACS+ server to recognize the CHAP vendor-specific attributes (VSAs).

Note Cisco Nexus® 3550-T switches support the CLI command, aaa authentication login ascii-authentication, only for TACAAS+, but not for RADIUS. Ensure that you have disabled aaa authentication login ascii-authentication switch so that the default authentication, PAP, is enabled. Otherwise, you will see syslog errors.

This table shows the RADIUS and TACACS+ VSAs required for CHAP.

Table 5: CHAP RADIUS and TACACS+ VSAs

Vendor-ID Number	Vendor-Type Number	VSA	Description
311	11	CHAP-Challenge	Contains the challenge sent by an AAA server to a CHAP user. It can be used in both Access-Request and Access-Challenge packets.
211	11	CHAP-Response	Contains the response value provided by a CHAP user in response to the challenge. It is used only in Access-Request packets.

Before you begin

Disable AAA ASCII authentication for logins.

Procedure

	Command or Action	Purpose

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

Step 2	**no aaa authentication login ascii-authentication** **Example:** `switch(config)# ` **`no aaa authentication login ascii-authentication`**	Disables ASCII authentication.
Step 3	**aaa authentication login chap enable** **Example:** `switch(config)# ` **`aaa authentication login chap enable`**	Enables CHAP authentication. The default is disabled. **Note** You cannot enable both CHAP and MSCHAP or MSCHAP V2 on your Cisco NX-OS device.

Step 4	(Optional) **exit**
	Example:
	`switch(config)# ` **`exit`**
	`switch#`
		Exits configuration mode.

Step 5	(Optional) **show aaa authentication login chap** **Example:** `switch# ` **`show aaa authentication login chap`**	Displays the CHAP configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch# ` **`copy running-config startup-config`**	Copies the running configuration to the startup configuration.

Enabling MSCHAP or MSCHAP V2 Authentication

Microsoft Challenge Handshake Authentication Protocol (MSCHAP) is the Microsoft version of CHAP. The Cisco NX-OS software also supports MSCHAP Version 2 (MSCHAP V2). You can use MSCHAP for user logins to a Cisco NX-OS device through a remote authentication server (RADIUS or TACACS+). MSCHAP V2 only supports user logins to a Cisco NX-OS device through remote authentication RADIUS servers. If you configure a TACACS+ group with MSCHAP V2, the AAA default login authentication uses the next configured method, or the local method, if no other server group is configured.

Note The Cisco NX-OS software may display the following message:

" Warning: MSCHAP V2 is supported only with Radius."

This warning message is informational only and does not affect MSCHAP V2 operation with RADIUS.

By default, the Cisco NX-OS device uses Password Authentication Protocol (PAP) authentication between the Cisco NX-OS device and the remote server. If you enable MSCHAP or MSCHAP V2, you need to configure your RADIUS server to recognize the MSCHAP and MSCHAP V2 vendor-specific attributes (VSAs).

This table shows the RADIUS VSAs required for MSCHAP.

Table 6: MSCHAP and MSCHAP V2 RADIUS VSAs

Vendor-ID Number	Vendor-Type Number	VSA	Description
311	11	MSCHAP-Challenge	Contains the challenge sent by an AAA server to an MSCHAP or MSCHAP V2 user. It can be used in both Access-Request and Access-Challenge packets.
211	11	MSCHAP-Response	Contains the response value provided by an MSCHAP or MSCHAP V2 user in response to the challenge. It is only used in Access-Request packets.

Before you begin

Disable AAA ASCII authentication for logins.

Procedure

	Command or Action	Purpose

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

Step 2	**no aaa authentication login ascii-authentication** **Example:** `switch(config)# no aaa authentication login ascii-authentication`	Disables ASCII authentication.
Step 3	**aaa authentication login** {**mschap** | **mschapv2**} **enable** **Example:** `switch(config)# aaa authentication login mschap enable`	Enables MSCHAP or MSCHAP V2 authentication. The default is disabled. **Note** You cannot enable both MSCHAP and MSCHAP V2 on your Cisco NX-OS device.

	Command or Action	Purpose

Step 4	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 5	(Optional) **show aaa authentication login** {**mschap** | **mschapv2**} **Example:** `switch# show aaa authentication login mschap`	Displays the MSCHAP or MSCHAP V2 configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring AAA Accounting Default Methods

Cisco NX-OS software supports TACACS+ and RADIUS methods for accounting. Cisco NX-OS devices report user activity to TACACS+ or RADIUS security servers in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the AAA server.

When you activate AAA accounting, the Cisco NX-OS device reports these attributes as accounting records, which are then stored in an accounting log on the security server.

You can create default method lists defining specific accounting methods, which include the following:

RADIUS server group
Uses the global pool of RADIUS servers for accounting.

Specified server group
Uses a specified RADIUS or TACACS+ server group for accounting.

Local
Uses the local username or password database for accounting.

Note If you have configured server groups and the server groups do not respond, by default, the local database is used for authentication.

Before you begin

Configure RADIUS or TACACS+ server groups, as needed.

Procedure

	Command or Action	Purpose

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

Step 2	**aaa accounting default** {**group** *group-list* | **local**}
	Example:
	`switch(config)# aaa accounting default group radius`
		Configures the default accounting method.
		The *group-list* argument consists of a space-delimited list of group names. The group names are the following:
		• **radius**—Uses the global pool of RADIUS servers for accounting.
		• *named-group*—Uses a named subset of TACACS+ or RADIUS servers for accounting.
		The **local** method uses the local database for accounting.
		The default method is **local**, which is used when no server groups are configured or when all the configured server groups fail to respond.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 4	(Optional) **show aaa accounting** **Example:** `switch# show aaa accounting`	Displays the configuration AAA accounting default methods.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Using AAA Server VSAs with Cisco NX-OS Devices

You can use vendor-specific attributes (VSAs) to specify Cisco NX-OS user roles and SNMPv3 parameters on AAA servers.

About VSAs

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating VSAs between the network access server and the RADIUS server. The IETF uses attribute 26. VSAs allow vendors to support their own extended attributes that are not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option using the format recommended in the specification. The Cisco vendor ID is 9, and the supported option is vendor type 1, which is named cisco-av-pair. The value is a string with the following format:

```
protocol : attribute separator value *
```

The protocol is a Cisco attribute for a particular type of authorization, the separator is = (equal sign) for mandatory attributes, and * (asterisk) indicates optional attributes.

When you use RADIUS servers for authentication on a Cisco NX-OS device, the RADIUS protocol directs the RADIUS server to return user attributes, such as authorization information, along with authentication results. This authorization information is specified through VSAs.

VSA Format

The following VSA protocol options are supported by the Cisco NX-OS software:

Shell

Protocol used in access-accept packets to provide user profile information.

Accounting

Protocol used in accounting-request packets. If a value contains any white spaces, put it within double quotation marks.

The following attributes are supported by the Cisco NX-OS software:

roles

Lists all the roles assigned to the user. The value field is a string that stores the list of group names delimited by white space. For example, if you belong to role network-operator and network-admin, the value field would be network-operator network-admin. This subattribute is sent in the VSA portion of the Access-Accept frames from the RADIUS server, and it can only be used with the shell protocol value. These examples use the roles attribute:

```
shell:roles=network-operator network-admin
```

```
shell:roles*network-operator network-admin
```

The following examples show the roles attribute as supported by FreeRADIUS:

```
Cisco-AVPair = shell:roles=\network-operator network-admin\
```

```
Cisco-AVPair = shell:roles*\network-operator network-admin\
```

Note When you specify a VSA as shell:roles*"network-operator network-admin" or "shell:roles*\"network-operator network-admin\"", this VSA is flagged as an optional attribute and other Cisco devices ignore this attribute.

accountinginfo

Stores accounting information in addition to the attributes covered by a standard RADIUS accounting protocol. This attribute is sent only in the VSA portion of the Account-Request frames from the RADIUS client on the switch, and it can only be used with the accounting protocol-related PDUs.

Specifying Cisco NX-OS User Roles and SNMPv3 Parameters on AAA Servers

You can use the VSA cisco-av-pair on AAA servers to specify user role mapping for the Cisco NX-OS device using this format:

```
shell:roles="roleA roleB …"
```

If you do not specify the role option in the cisco-av-pair attribute, the default user role is network-operator.

You can also specify your SNMPv3 authentication and privacy protocol attributes as follows:

```
shell:roles="roleA roleB..." snmpv3:auth=SHA priv=AES-128
```

The SNMPv3 authentication protocol options are SHA and MD5. The privacy protocol options are AES-128 and DES. If you do not specify these options in the cisco-av-pair attribute, MD5 and DES are the default authentication protocols.

Configuring Secure Login Features

Configuring Login Parameters

You can configure login parameters to automatically block further login attempts when a possible denial-of-service (DoS) attack is detected and slow down dictionary attacks by enforcing a quiet period if multiple failed connection attempts are detected.

Note This feature restarts if a system switchover occurs or the AAA process restarts.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	[**no**] **login block-for** *seconds* **attempts** *tries* **within** *seconds* **Example:** `switch(config)# login block-for 100 attempts 2 within 60`	Configures the quiet mode time period. The range for all arguments is from 1 to 65535. The example shows how to configure the switch to enter a 100-second quiet period if 2 failed login attempts are exceeded within 60 seconds. After you enter this command, all login attempts made through Telnet or SSH are denied during the quiet period. Access control lists (ACLs)

	Command or Action	Purpose
		are not exempt from the quiet period until the command is entered. **Note** You must enter this command before any other login command can be used.
Step 3	(Optional) [**no**] **login quiet-mode access-class** *acl-name* **Example:** `switch(config)#  login quiet-mode access-class myacl`	Specifies an ACL that is to be applied to the switch when it changes to quiet mode. When the switch is in quiet mode, all login requests are denied, and the only available connection is through the console.
Step 4	(Optional) **show login** [**failures**] **Example:** `switch(config)# show  login`	Displays the login parameters. The **failures** option displays information related only to failed login attempts.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Restricting User Login Sessions

You can restrict the maximum number of simultaneous login sessions per user. Doing so prevents users from having multiple unwanted sessions and solves the potential security issue of unauthorized users accessing a valid SSH or Telnet session.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	[**no**] **user max-logins** *max-logins* **Example:** `switch(config)# user max-logins 1`	Restricts the maximum number of simultaneous login sessions per user. The range is from 1 to 7. If you set the maximum login limit as 1, only one Telnet or SSH session is allowed per user. **Note** The configured login limit applies to all users. You cannot set a different limit for individual users.
Step 3	(Optional) **show running-config all | i max-login** **Example:**	Displays the maximum number of login sessions allowed per user.

	Command or Action	Purpose
	`switch(config)# show running-config all | i max-login`	
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Restricting the Password Length

You can restrict the minimum and maximum length of the user password. This feature enables you to increase system security by forcing the user to provide a strong password.

Before you begin

You must enable password strength checking using the **password strength-check** command. If you restrict the password length but do not enable password strength checking and the user enters a password that is not within the restricted length, an error appears, but a user account is created. To enforce the password length and prevent a user account from being created, you must enable password strength checking and restrict the password length.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.

Step 2	[**no**] **userpassphrase** {**min-length** *min-length* | **max-length** *max-length*}
	Example:
	`switch(config)# userpassphrase min-length 8 max-length 80`
		Restricts the minimum and/or maximum length of the user password. The minimum password length is from 4 to 127 characters, and the maximum password length is from 80 to 127 characters.

Step 3	(Optional) **show userpassphrase** {**length** | **max-length** | **min-length**}
	Example:
	`switch(config)# show userpassphrase length`
		Displays the minimum and maximum length of the user password.

Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Enabling the Password Prompt for the Username

You can configure the switch to prompt the user to enter a password after entering the username.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.

Step 2	**password prompt username**
	Example:
	`switch(config)# password prompt username`
	`Password prompt username is enabled.`
	`After providing the required options in`
	` the username command, press enter.`
	`User will be prompted for the username`
	`password and password will be hidden.`
	`Note: Choosing password key in the same`
	` line while configuring user account,`
	`password will not be hidden.`
		Configures the switch to prompt the user to enter a password after she enters the **username** command without the **password** option or the **snmp-server user** command. The password that the user enters will be hidden. You can use the **no** form of this command to disable this feature.

Step 3	(Optional) **copy running-config startup-config**
	Example:
	`switch(config)# copy running-config`
	`startup-config`
		Copies the running configuration to the startup configuration.

Configuring the Shared Secret for RADIUS or TACACS+

The shared secret that you configure for remote authentication and accounting between the switch and the RADIUS or TACACS+ server should be hidden because it is sensitive information. You can use a separate command to generate an encrypted shared secret for the **radius-server** [**host**] **key** and **tacacs-server** [**host**] **key** commands. The SHA256 hashing method is used to store the encrypted shared secret.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.

Step 2	**generate type7_encrypted_secret**
	Example:
	`switch(config)# generate`
	`type7_encrypted_secret`
	`Type-7 (Vigenere) Encryption,`
	` Use this encrypted secret to configure`
		Configures the RADIUS or TACACS+ shared secret with key type 7. You are prompted to enter the shared secret in plain text twice. The secret is hidden as you enter it. Then an encrypted version of the secret appears.

	Command or Action	Purpose

	` radius and tacacs shared secret with key type 7.`
	` Copy complete secret with double quotes.`
	`Enter plain text secret:`
	`Confirm plain text secret:`
	`Type 7 Encrypted secret is : "fewhg"`
		Note You can generate the encrypted equivalent of a plain-text secret separately and configure the encrypted shared secret later using the **radius-server** [**host**] **key** and **tacacs-server** [**host**] **key** commands.

Step 3	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Monitoring and Clearing the Local AAA Accounting Log

The Cisco NX-OS device maintains a local log for the AAA accounting activity. You can monitor this log and clear it.

Procedure

	Command or Action	Purpose
Step 1	**show accounting log** [*size* | **last-index** | **start-seqnum** *number* | **start-time** *year month day hh***:***mm***:***ss*] **Example:** `switch# show accounting log`	Displays the accounting log contents. By default, the command output contains up to 250,000 bytes of the accounting log. You can use the *size* argument to limit command output. The range is from 0 to 250000 bytes. You can also specify a starting sequence number or a starting time for the log output.The range of the starting index is from 1 to 1000000. Use the **last-index** keyword to display the value of the last index number in the accounting log file.
Step 2	(Optional) **clear accounting log** [**logflash**] **Example:** `switch# clear aaa accounting log`	Clears the accounting log contents. The **logflash** keyword clears the accounting log stored in the logflash.

Verifying the AAA Configuration

To display AAA configuration information, perform one of the following tasks:

Command	Purpose
show aaa accounting	Displays AAA accounting configuration.

Command	Purpose
show aaa authentication [**login** {**ascii-authentication** | **chap** | **error-enable** | **mschap** | **mschapv2**}]	Displays AAA authentication login configuration information.
show aaa groups	Displays the AAA server group configuration.
show login [**failures**]	Displays the login parameters. The **failures** option displays information related only to failed login attempts. **Note** The **clear login failures** command clears the login failures in the current watch period.
show login on-failure log	Displays whether the switch is configured to log failed authentication messages to the syslog server.
show login on-successful log	Displays whether the switch is configured to log successful authentication messages to the syslog server.
show running-config aaa [**all**]	Displays the AAA configuration in the running configuration.
show running-config all | i max-login	Displays the maximum number of login sessions allowed per user.
show startup-config aaa	Displays the AAA configuration in the startup configuration.
show userpassphrase {**length** | **max-length** | **min-length**}	Displays the minimum and maximum length of the user password.

Configuration Examples for AAA

The following example shows how to configure AAA:

```
aaa authentication login default group radius
aaa authentication login console group radius
aaa accounting default group radius
```

Configuration Examples for Login Parameters

The following example shows how to configure the switch to enter a 100-second quiet period if 3 failed login attempts is exceeded within 60 seconds. This example shows no login failures.

```
switch# configure terminal
switch(config)# login block-for 100 attempts 3 within 60
switch(config)# show login

No Quiet-Mode access list has been configured, default ACL will be applied.

 Switch is enabled to watch for login Attacks.
 If more than 3 login failures occur in 60 seconds or less,
 logins will be disabled for 100 seconds.

 Switch presently in Normal-Mode.
 Current Watch Window remaining time 45 seconds.
 Present login failure count 0.

switch(config)# show login failures
*** No logged failed login attempts with the device.***
```

The following example shows how to configure a quiet-mode ACL. All login requests are denied during the quiet period except hosts from the myacl ACL. This example also shows a login failure.

```
switch# configure terminal
switch(config)# login block-for 100 attempts 3 within 60
switch(config)# login quiet-mode access-class myacl

switch(config)# show login

 Switch is enabled to watch for login Attacks.
 If more than 3 login failures occur in 60 seconds or less,
 logins will be disabled for 100 seconds.

 Switch presently in Quiet-Mode.
 Will remain in Quiet-Mode for 98 seconds.
 Denying logins from all sources.

switch(config)# show login failures
Information about last 20 login failure's with the device.
--------------------------------------------------------------------------------
Username      Line          SourceIPAddr    Appname    TimeStamp
--------------------------------------------------------------------------------
asd           /dev/pts/0    171.70.55.158   login      Mon Aug  3 18:18:54 2015
qweq          /dev/pts/0    171.70.55.158   login      Mon Aug  3 18:19:02 2015
qwe           /dev/pts/0    171.70.55.158   login      Mon Aug  3 18:19:08 2015
--------------------------------------------------------------------------------
```

Configuration Examples for the Password Prompt Feature

The following example shows how to configure the switch to prompt the user to enter a password after she enters the **username** command and the error message that displays if she does not enter a password.

```
switch# configure terminal
switch(config)# password prompt username
Password prompt username is enabled.
```

```
After providing the required options in the username command, press enter.
User will be prompted for the username password and password will be hidden.
Note: Choosing password key in the same line while configuring user account, password will
 not be hidden.


switch(config)# username user1
Enter password:
Confirm password:
warning: password for user:user1 not set. S/he may not be able to login
```

The following example shows how to configure the switch to prompt the user to enter a password after she enters the **snmp-server user** command and the prompts that then display to the user.

```
switch# configure terminal
switch(config)# password prompt username
Password prompt username is enabled.
After providing the required options in the username command, press enter.
User will be prompted for the username password and password will be hidden.
Note: Choosing password key in the same line while configuring user account, password will
 not be hidden.


N3550-T(config)# snmp-server user user1
Enter auth md5 password (Press Enter to Skip):
Enter auth sha password (Press Enter to Skip):
```

Additional References for AAA

This section includes additional information related to implementing AAA.

Related Documents

Related Topic	Document Title
Cisco NX-OS Licensing	*Cisco NX-OS Licensing Guide*

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
MIBs related to AAA	To locate and download supported MIBs, go to the following URL: ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

CHAPTER 4

Configuring RADIUS

This chapter describes how to configure the Remote Access Dial-In User Service (RADIUS) protocol on Cisco NX-OS devices.

This chapter includes the following sections:

- About RADIUS, on page 37
- About RADIUS Change of Authorization, on page 40
- Prerequisites for RADIUS, on page 41
- Guidelines and Limitations for RADIUS, on page 41
- Default Settings for RADIUS, on page 42
- Configuring RADIUS Servers, on page 42
- Enabling or Disabling Dynamic Author Server, on page 59
- Configuring RADIUS Change of Authorization, on page 60
- Verifying the RADIUS Configuration, on page 60
- Verifying RADIUS Change of Authorization Configuration, on page 61
- Monitoring RADIUS Servers, on page 61
- Clearing RADIUS Server Statistics, on page 62
- Configuration Example for RADIUS, on page 62
- Configuration Examples of RADIUS Change of Authorization, on page 63
- Additional References for RADIUS, on page 63

About RADIUS

The RADIUS distributed client/server system allows you to secure networks against unauthorized access. In the Cisco implementation, RADIUS clients run on Cisco NX-OS devices and send authentication and accounting requests to a central RADIUS server that contains all user authentication and network service access information.

RADIUS Network Environments

RADIUS can be implemented in a variety of network environments that require high levels of security while maintaining network access for remote users.

You can use RADIUS in the following network environments that require access security:

- Networks with multiple-vendor network devices, each supporting RADIUS. For example, network devices from several vendors can use a single RADIUS server-based security database.

- Networks already using RADIUS. You can add a Cisco NX-OS device with RADIUS to the network. This action might be the first step when you make a transition to a AAA server.
- Networks that require resource accounting. You can use RADIUS accounting independent of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, indicating the amount of resources (such as time, packets, bytes, and so on) used during the session. An Internet service provider (ISP) might use a freeware-based version of the RADIUS access control and accounting software to meet special security and billing needs.
- Networks that support authentication profiles. Using the RADIUS server in your network, you can configure AAA authentication and set up per-user profiles. Per-user profiles enable the Cisco NX-OS device to better manage ports using their existing RADIUS solutions and to efficiently manage shared resources to offer different service-level agreements.

RADIUS Operation

When a user attempts to log in and authenticate to a Cisco NX-OS device using RADIUS, the following process occurs:

- The user is prompted for and enters a username and password.
- The username and encrypted password are sent over the network to the RADIUS server.
- The user receives one of the following responses from the RADIUS server:

  ACCEPT
  The user is authenticated.
  REJECT
  The user is not authenticated and is prompted to reenter the username and password, or access is denied.
  CHALLENGE
  A challenge is issued by the RADIUS server. The challenge collects additional data from the user.
  CHANGE PASSWORD
  A request is issued by the RADIUS server, asking the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for EXEC or network authorization. You must first complete RADIUS authentication before using RADIUS authorization. The additional data included with the ACCEPT or REJECT packets consists of the following:

- Services that the user can access, including Telnet, rlogin, or local-area transport (LAT) connections, and Point-to-Point Protocol (PPP), Serial Line Internet Protocol (SLIP), or EXEC services.
- Connection parameters, including the host or client IPv4 address, access list, and user timeouts.

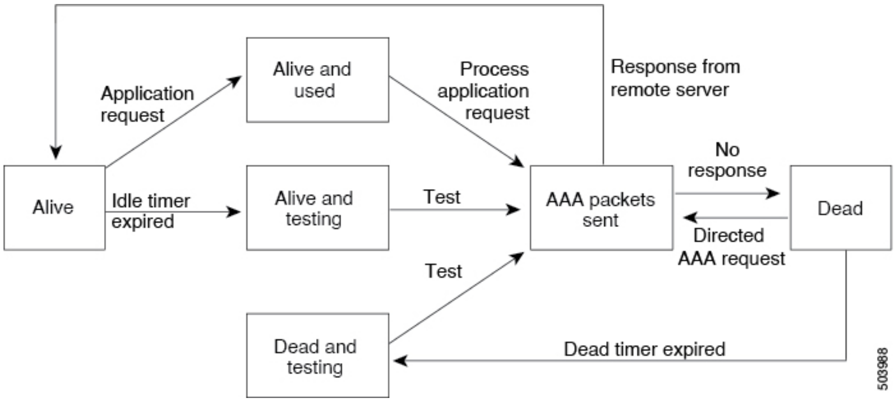
RADIUS Server Monitoring

An unresponsive RADIUS server can cause a delay in processing AAA requests. You can configure the Cisco NX-OS device to periodically monitor a RADIUS server to check whether it is responding (or alive) to save time in processing AAA requests. The Cisco NX-OS device marks unresponsive RADIUS servers as dead and does not send AAA requests to any dead RADIUS servers. The Cisco NX-OS device periodically monitors the dead RADIUS servers and brings them to the alive state once they respond. This monitoring process verifies that a RADIUS server is in a working state before real AAA requests are sent its way. Whenever a

RADIUS server changes to the dead or alive state, a Simple Network Management Protocol (SNMP) trap is generated and the Cisco NX-OS device displays an error message that a failure is taking place.

Figure 1: RADIUS Server States

This figure shows the states for RADIUS server monitoring.



Note The monitoring interval for alive servers and dead servers are different and can be configured by the user. The RADIUS server monitoring is performed by sending a test authentication request to the RADIUS server.

Vendor-Specific Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating VSAs between the network access server and the RADIUS server. The IETF uses attribute 26. VSAs allow vendors to support their own extended attributes that are not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option using the format recommended in the specification. The Cisco vendor ID is 9, and the supported option is vendor type 1, which is named cisco-av-pair. The value is a string with the following format:

```
protocol : attribute separator value *
```

The protocol is a Cisco attribute for a particular type of authorization, the separator is = (equal sign) for mandatory attributes, and * (asterisk) indicates optional attributes.

When you use RADIUS servers for authentication on a Cisco NX-OS device, the RADIUS protocol directs the RADIUS server to return user attributes, such as authorization information, with authentication results. This authorization information is specified through VSAs.

The following VSA protocol options are supported by the Cisco NX-OS software:

Shell
Protocol used in access-accept packets to provide user profile information.

Accounting
Protocol used in accounting-request packets. If a value contains any white spaces, you should enclose the value within double quotation marks.

The Cisco NX-OS software supports the following attributes:

roles

Lists all the roles to which the user belongs. The value field is a string that lists the role names delimited by white space. For example, if the user belongs to roles network-operator and network-admin, the value field would be network-operator network-admin. This subattribute, which the RADIUS server sends in the VSA portion of the Access-Accept frames, can only be used with the shell protocol value. The following examples show the roles attribute that is supported by the Cisco Access Control Server (ACS):

```
shell:roles=network-operator network-admin
```

```
shell:roles*“network-operator network-admin
```

The following examples show the roles attribute that is supported by FreeRADIUS:

```
Cisco-AVPair = shell:roles=\network-operator network-admin\
```

```
Cisco-AVPair = shell:roles*\network-operator network-admin\
```

Note When you specify a VSA as shell:roles*"network-operator network-admin" or "shell:roles*\"network-operator network-admin\"", this VSA is flagged as an optional attribute and other Cisco devices ignore this attribute.

accountinginfo

Stores accounting information in addition to the attributes covered by a standard RADIUS accounting protocol. This attribute is sent only in the VSA portion of the Account-Request frames from the RADIUS client on the switch. It can be used only with the accounting protocol data units (PDUs).

About RADIUS Change of Authorization

A standard RADIUS interface is typically used in a pulled model, in which the request originates from a device attached to a network and the response is sent from the queried servers. Cisco NX-OS sofware supports the RADIUS Change of Authorization (CoA) request defined in RFC 5176 that is used in a pushed model, in which the request originates from the external server to the device attached to the network, and enables the dynamic reconfiguring of sessions from external authentication, authorization, and accounting (AAA) or policy servers.

When Dot1x is enabled, the network device acts as the authenticator and is responsible for processing dynamic COA per session.

The following requests are supported:

- Session reauthentication
- Session termination

Session Reauthentication

To initiate session reauthentication, the authentication, authorization, and accounting (AAA) server sends a standard CoA-Request message that contains a Cisco VSA and one or more session identification attributes. The Cisco VSA is in the form of Cisco:Avpair="subscriber:command=reauthenticate".

The current session state determines the response of the device to the message in the following scenarios:

- If the session is currently authenticated by IEEE 802.1x, the device responds by sending an Extensible Authentication Protocol over LAN (EAPOL)-RequestId message to the server.
- If the session is currently authenticated by MAC authentication bypass (MAB), the device sends an access request to the server, passing the same identity attributes used for the initial successful authentication.
- If session authentication is in progress when the device receives the command, the device terminates the process and restarts the authentication sequence, starting with the method configured to be attempted first.

Session Termination

A CoA Disconnect-Request terminates the session without disabling the host port. CoA Disconnect-Request termination causes reinitialization of the authenticator state machine for the specified host, but does not restrict the host's access to the network.

If the session cannot be located, the device returns a Disconnect-NAK message with the "Session Context Not Found" error-code attribute.

If the session is located, but the NAS was unable to remove the session due to some internal error, the device returns a Disconnect-NAK message with the "Session Context Not Removable" error-code attribute.

If the session is located, the device terminates the session. After the session has been completely removed, the device returns a Disconnect-ACK message.

Prerequisites for RADIUS

RADIUS has the following prerequisites:

- Obtain IPv4 addresses or hostnames for the RADIUS servers.
- Obtain keys from the RADIUS servers.
- Ensure that the Cisco NX-OS device is configured as a RADIUS client of the AAA servers.

Guidelines and Limitations for RADIUS

RADIUS has the following guidelines and limitations:

- You can configure a maximum of 64 RADIUS servers on the Cisco NX-OS device.

- If you have a user account configured on the local Cisco NX-OS device that has the same name as a remote user account on an AAA server, the Cisco NX-OS software applies the user roles for the local user account to the remote user, not the user roles configured on the AAA server.
- Only the RADIUS protocol supports one-time passwords.
- Cisco Nexus® 3550-T switches support the CLI command, aaa authentication login ascii-authentication, only for TACAAS+, but not for RADIUS. Ensure that you have disabled aaa authentication login ascii-authentication switch so that the default authentication, PAP, is enabled. Otherwise, you will see syslog errors.

Default Settings for RADIUS

This table lists the default settings for RADIUS parameters.

Table 7: Default RADIUS Parameter Settings

Parameters	Default
Server roles	Authentication and accounting
Dead timer interval	0 minutes
Retransmission count	1
Retransmission timer interval	5 seconds
Authentication port	1812
Accounting port	1813
Idle timer interval	0 minutes
Periodic server monitoring username	test
Periodic server monitoring password	test

Configuring RADIUS Servers

This section describes how to configure RADIUS servers on a Cisco NX-OS device.

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Note Cisco Nexus® 3550-T switches support the CLI command, aaa authentication login ascii-authentication, only for TACAAS+, but not for RADIUS. Ensure that you have disabled aaa authentication login ascii-authentication switch so that the default authentication, PAP, is enabled. Otherwise, you will see syslog errors.

RADIUS Server Configuration Process

1. Establish the RADIUS server connections to the Cisco NX-OS device.
2. Configure the RADIUS secret keys for the RADIUS servers.
3. If needed, configure RADIUS server groups with subsets of the RADIUS servers for AAA authentication methods.
4. If needed, configure any of the following optional parameters:
   - Dead-time interval
   - RADIUS server specification allowed at user login
   - Timeout interval
   - TCP port
5. (Optional) If RADIUS distribution is enabled, commit the RADIUS configuration to the fabric.

Related Topics

Configuring RADIUS Server Hosts, on page 43
Configuring Global RADIUS Keys, on page 44

Configuring RADIUS Server Hosts

To access a remote RADIUS server, you must configure the IP address or hostname of a RADIUS server. You can configure up to 64 RADIUS servers.

Note By default, when you configure a RADIUS server IP address or hostname of the Cisco NX-OS device, the RADIUS server is added to the default RADIUS server group. You can also add the RADIUS server to another RADIUS server group.

Before you begin

Ensure that the server is already configured as a member of the server group.

Ensure that the server is configured to authenticate RADIUS traffic.

Ensure that the Cisco NX-OS device is configured as a RADIUS client of the AAA servers.

Procedure

	Command or Action	Purpose

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

Step 2	**radius-server host** {*ipv4-address* | | *hostname*}
	Example:
	`switch(config)# radius-server host 10.10.1.1`
		Specifies the IPv4 address or hostname for a RADIUS server to use for authentication.

Step 3	(Optional) **show radius** {**pending** | **pending-diff**}
	Example:
	`switch(config)# show radius pending`
		Displays the RADIUS configuration pending for distribution.

Step 4	(Optional) **radius commit** **Example:** `switch(config)# radius commit`	Applies the RADIUS configuration changes in the temporary database to the running configuration.

Step 5	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 6	(Optional) **show radius-server** **Example:** `switch# show radius-server`	Displays the RADIUS server configuration.
Step 7	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring a Key for a Specific RADIUS Server, on page 45

Configuring Global RADIUS Keys

You can configure RADIUS keys for all servers used by the Cisco NX-OS device. A RADIUS key is a shared secret text string between the Cisco NX-OS device and the RADIUS server hosts.

Before you begin

Obtain the RADIUS key values for the remote RADIUS servers.

Configure the RADIUS key on the remote RADIUS servers.

Procedure

	Command or Action	Purpose

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

Step 2	**radius-server key** [**0** | **6** | **7**] *key-value* **Example:** `switch(config)# radius-server key 0 QsEfThUkO`	Specifies a RADIUS key for all RADIUS servers. You can specify that the *key-value* is in clear text format (**0**), is type-6 encrypted (**6**), or is type-7 encrypted (**7**). The Cisco NX-OS software encrypts a clear text key before saving it to the running configuration. The default format is clear text. The maximum length is 63 characters. By default, no RADIUS key is configured.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 4	(Optional) **show radius-server** **Example:** `switch# show radius-server`	Displays the RADIUS server configuration. **Note** The RADIUS keys are saved in encrypted form in the running configuration. Use the **show running-config** command to display the encrypted RADIUS keys.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring RADIUS Server Groups, on page 47

Configuring a Key for a Specific RADIUS Server

You can configure a key on the Cisco NX-OS device for a specific RADIUS server. A RADIUS key is a secret text string shared between the Cisco NX-OS device and a specific RADIUS server.

Before you begin

Configure one or more RADIUS server hosts.

Obtain the key value for the remote RADIUS server.

Configure the key on the RADIUS server.

Procedure

	Command or Action	Purpose

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

Step 2	**radius-server host** {*ipv4-address* | *hostname*} **key** [**0** | **6** | **7**] *key-value* **Example:** `switch(config)# radius-server host 10.10.1.1 key 0 PlIjUhYg`	Specifies a RADIUS key for a specific RADIUS server. You can specify that the *key-value* is in clear text format (**0**), is type-6 encrypted (**6**), or is type-7 encrypted (**7**). The Cisco NX-OS software encrypts a clear text key before saving it to the running configuration. The default format is clear text. The maximum length is 63 characters. This RADIUS key is used instead of the global RADIUS key.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 4	(Optional) **show radius-server** **Example:** `switch# show radius-server`	Displays the RADIUS server configuration. **Note** The RADIUS keys are saved in encrypted form in the running configuration. Use the **show running-config** command to display the encrypted RADIUS keys.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring RADIUS Server Hosts, on page 43

Configuring RADIUS Server Groups

You can specify one or more remote AAA servers for authentication using server groups. All members of a group must belong to the RADIUS protocol. The servers are tried in the same order in which you configure them.

You can configure these server groups at any time but they only take effect when you apply them to an AAA service.

Before you begin

Ensure that all servers in the group are RADIUS servers.

Procedure

	Command or Action	Purpose

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

Step 2	**aaa group server radius** *group-name*
	Example:
	`switch(config)# aaa group server radius RadServer`
	`switch(config-radius)#`
		Creates a RADIUS server group and enters the RADIUS server group configuration submode for that group. The *group-name* argument is a case-sensitive alphanumeric string with a maximum length of 127 characters.
		To delete a RADIUS server group, use the **no** form of this command.
		Note You are not allowed to delete the default system generated default group (RADIUS).

Step 3	**server** {*ipv4-address* | *hostname*}
	Example:
	`switch(config-radius)# server 10.10.1.1`
		Configures the RADIUS server as a member of the RADIUS server group.
		If the specified RADIUS server is not found, configure it using the **radius-server host** command and retry this command.

Step 4	(Optional) **deadtime** *minutes* **Example:** `switch(config-radius)# deadtime 30`	Configures the monitoring dead time. The default is 0 minutes. The range is from 1 through 1440. **Note** If the dead-time interval for a RADIUS server group is greater than zero (0), that value takes precedence over the global dead-time value.
Step 5	(Optional) **server** {*ipv4-address* | *hostname*} **Example:**	Configures the RADIUS server as a member of the RADIUS server group.

	Command or Action	Purpose
	`switch(config-radius)# server 10.10.1.1`	**Tip** If the specified RADIUS server is not found, configure it using the **radius-server host** command and retry this command.
Step 6	(Optional) **use-vrf** *vrf-name* **Example:** `switch(config-radius)# use-vrf vrf1`	Specifies the VRF to use to contact the servers in the server group.

Step 7	**exit**
	Example:
	`switch(config-radius)# exit`
	`switch(config)#`
		Exits configuration mode.

Step 8	(Optional) **show radius-server groups** [*group-name*] **Example:** `switch(config)# show radius-server groups`	Displays the RADIUS server group configuration.
Step 9	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring the RADIUS Dead-Time Interval, on page 57

Configuring the Global Source Interface for RADIUS Server Groups

You can configure a global source interface for RADIUS server groups to use when accessing RADIUS servers. You can also configure a different source interface for a specific RADIUS server group. By default, the Cisco NX-OS software uses any available interface.

Procedure

	Command or Action	Purpose

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

Step 2	**ip radius source-interface** *interface* **Example:** `switch(config)# ip radius source-interface mgmt 0`	Configures the global source interface for all RADIUS server groups configured on the device.

	Command or Action	Purpose

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 4	(Optional) **show radius-server** **Example:** `switch# show radius-server`	Displays the RADIUS server configuration information.
Step 5	(Optional) **copy running-config startup config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring RADIUS Server Groups, on page 47

Allowing Users to Specify a RADIUS Server at Login

By default, the Cisco NX-OS device forwards an authentication request based on the default AAA authentication method. You can configure the Cisco NX-OS device to allow the user to specify a RADIUS server to send the authentication request by enabling the directed-request option. If you enable this option, the user can log in as *username*@**vrfname***hostname*, where **hostname** is the VRF to use and **hostname** is the name of a configured RADIUS server.

Note If you enable the directed-request option, the Cisco NX-OS device uses only the RADIUS method for authentication and not the default local method.

Note User-specified logins are supported only for Telnet sessions.

Procedure

	Command or Action	Purpose

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

	Command or Action	Purpose
Step 2	**radius-server directed-request** **Example:** `switch(config)# radius-server directed-request`	Allows users to specify a RADIUS server to send the authentication request when logging in. The default is disabled.

Step 3	(Optional) **show radius** {**pending** | **pending-diff**}
	Example:
	`switch(config)# show radius pending`
		Displays the RADIUS configuration pending for distribution.

Step 4	(Optional) **radius commit** **Example:** `switch(config)# radius commit`	Applies the RADIUS configuration changes in the temporary database to the running configuration.

Step 5	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 6	(Optional) **show radius-server directed-request** **Example:** `switch# show radius-server directed-request`	Displays the directed request configuration.
Step 7	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring the Global RADIUS Transmission Retry Count and Timeout Interval

You can configure a global retransmission retry count and timeout interval for all RADIUS servers. By default, a Cisco NX-OS device retries transmission to a RADIUS server only once before reverting to local authentication. You can increase this number up to a maximum of five retries per server. The timeout interval determines how long the Cisco NX-OS device waits for responses from RADIUS servers before declaring a timeout failure.

Procedure

	Command or Action	Purpose

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

	Command or Action	Purpose
Step 2	**radius-server retransmit** *count* **Example:** `switch(config)# radius-server retransmit 3`	Specifies the retransmission count for all RADIUS servers. The default retransmission count is 1 and the range is from 0 to 5.
Step 3	**radius-server timeout** *seconds* **Example:** `switch(config)# radius-server timeout 10`	Specifies the transmission timeout interval for RADIUS servers. The default timeout interval is 5 seconds and the range is from 1 to 60 seconds.

Step 4	(Optional) **show radius** {**pending** | **pending-diff**}
	Example:
	`switch(config)# show radius pending`
		Displays the RADIUS configuration pending for distribution.

Step 5	(Optional) **radius commit** **Example:** `switch(config)# radius commit`	Applies the RADIUS configuration changes in the temporary database to the running configuration.

Step 6	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 7	(Optional) **show radius-server** **Example:** `switch# show radius-server`	Displays the RADIUS server configuration.
Step 8	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring the RADIUS Transmission Retry Count and Timeout Interval for a Server

By default, a Cisco NX-OS device retries a transmission to a RADIUS server only once before reverting to local authentication. You can increase this number up to a maximum of five retries per server. You can also set a timeout interval that the Cisco NX-OS device waits for responses from RADIUS servers before declaring a timeout failure.

Before you begin

Configure one or more RADIUS server hosts.

Procedure

	Command or Action	Purpose

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

Step 2	**radius-server host** {*ipv4-address* | *hostname*} **retransmit** *count* **Example:** `switch(config)# radius-server host server1 retransmit 3`	Specifies the retransmission count for a specific server. The default is the global value. **Note** The retransmission count value specified for a RADIUS server overrides the count specified for all RADIUS servers.
Step 3	**radius-server host** {*ipv4-address* | *hostname*} **timeout** *seconds* **Example:** `switch(config)# radius-server host server1 timeout 10`	Specifies the transmission timeout interval for a specific server. The default is the global value. **Note** The timeout interval value specified for a RADIUS server overrides the interval value specified for all RADIUS servers.

Step 4	(Optional) **show radius** {**pending** | **pending-diff**}
	Example:
	`switch(config)# show radius pending`
		Displays the RADIUS configuration pending for distribution.

Step 5	(Optional) **radius commit** **Example:** `switch(config)# radius commit`	Applies the RADIUS configuration changes in the temporary database to the running configuration and distributes RADIUS configuration to other Cisco NX-OS devices if you have enabled CFS configuration distribution for the user role feature.

Step 6	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 7	(Optional) **show radius-server** **Example:** `switch# show radius-server`	Displays the RADIUS server configuration.
Step 8	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring RADIUS Server Hosts, on page 43

Configuring Accounting and Authentication Attributes for RADIUS Servers

You can specify that a RADIUS server is to be used only for accounting purposes or only for authentication purposes. By default, RADIUS servers are used for both accounting and authentication. You can also specify the destination UDP port numbers where RADIUS accounting and authentication messages should be sent if there is a conflict with the default port.

Before you begin

Configure one or more RADIUS server hosts.

Procedure

	Command or Action	Purpose

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

Step 2	(Optional) **radius-server host** {*ipv4-address* | *hostname*} **acct-port** *udp-port* **Example:** `switch(config)# radius-server host 10.10.1.1 acct-port 2004`	Specifies a UDP port to use for RADIUS accounting messages. The default UDP port is 1813. The range is from 0 to 65535.
Step 3	(Optional) **radius-server host** {*ipv4-address* | *hostname*} **accounting** **Example:** `switch(config)# radius-server host 10.10.1.1 accounting`	Specifies to use the RADIUS server only for accounting purposes. The default is both accounting and authentication.
Step 4	(Optional) **radius-server host** {*ipv4-address* | *hostname*} **auth-port** *udp-port* **Example:** `switch(config)# radius-server host 10.10.2.2 auth-port 2005`	Specifies a UDP port to use for RADIUS authentication messages. The default UDP port is 1812. The range is from 0 to 65535.
Step 5	(Optional) **radius-server host** {*ipv4-address* | *hostname*} **authentication** **Example:** `switch(config)# radius-server host 10.10.2.2 authentication`	Specifies to use the RADIUS server only for authentication purposes. The default is both accounting and authentication.

	Command or Action	Purpose

Step 6	(Optional) **show radius** {**pending** | **pending-diff**}
	Example:
	`switch(config)# show radius pending`
		Displays the RADIUS configuration pending for distribution.

Step 7	(Optional) **radius commit** **Example:** `switch(config)# radius commit`	Applies the RADIUS configuration changes in the temporary database to the running configuration.

Step 8	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 9	(Optional) **show radius-server** **Example:** `switch(config)# show radius-server`	Displays the RADIUS server configuration.
Step 10	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring RADIUS Server Hosts, on page 43

Configuring Global Periodic RADIUS Server Monitoring

You can monitor the availability of all RADIUS servers without having to configure the test parameters for each server individually. Any servers for which test parameters are not configured are monitored using the global level parameters.

Note Test parameters that are configured for individual servers take precedence over global test parameters.

The global configuration parameters include the username and password to use for the servers and an idle timer. The idle timer specifies the interval in which a RADIUS server receives no requests before the Cisco NX-OS device sends out a test packet. You can configure this option to test servers periodically, or you can run a one-time only test.

Note To protect network security, we recommend that you use a username that is not the same as an existing username in the RADIUS database.

Note The default idle timer value is 0 minutes. When the idle time interval is 0 minutes, periodic RADIUS server monitoring is not performed.

Before you begin

Enable RADIUS.

Procedure

	Command or Action	Purpose

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

Step 2	**radius-server test** {**idle-time** *minutes* | **password** *password* [**idle-time** *minutes*] | **username** *name* [**password** *password* [**idle-time** *minutes*]]}
	Example:
	`switch(config)# `**`radius-server test username user1 password Ur2Gd2BH idle-time 3`**
		Specifies parameters for global server monitoring. The default username is test, and the default password is test. The default value for the idle timer is 0 minutes, and the valid range is from 0 to 1440 minutes.
		Note For periodic RADIUS server monitoring, the idle timer value must be greater than 0.

Step 3	**radius-server deadtime** *minutes* **Example:** `switch(config)# `**`radius-server deadtime 5`**	Specifies the number of minutes before the Cisco NX-OS device checks a RADIUS server that was previously unresponsive. The default value is 0 minutes, and the valid range is from 0 to 1440 minutes.

Step 4	**exit**
	Example:
	`switch(config)# `**`exit`**
	`switch#`
		Exits configuration mode.

Step 5	(Optional) **show radius-server** **Example:** `switch# `**`show radius-server`**	Displays the RADIUS server configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch# `**`copy running-config startup-config`**	Copies the running configuration to the startup configuration.

Related Topics

Configuring Periodic RADIUS Server Monitoring on Individual Servers, on page 56

Configuring Periodic RADIUS Server Monitoring on Individual Servers

You can monitor the availability of individual RADIUS servers. The configuration parameters include the username and password to use for the server and an idle timer. The idle timer specifies the interval during which a RADIUS server receives no requests before the Cisco NX-OS device sends out a test packet. You can configure this option to test servers periodically, or you can run a one-time only test.

Note Test parameters that are configured for individual servers take precedence over global test parameters.

Note For security reasons, we recommend that you do not configure a test username that is the same as an existing user in the RADIUS database.

Note The default idle timer value is 0 minutes. When the idle time interval is 0 minutes, the Cisco NX-OS device does not perform periodic RADIUS server monitoring.

Before you begin

Enable RADIUS.

Add one or more RADIUS server hosts.

Procedure

	Command or Action	Purpose

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

Step 2	**radius-server host** {*ipv4-address* | *hostname*} **test** {**idle-time** *minutes* | **password** *password* [**idle-time** *minutes*] | **username** *name* [**password** *password* [**idle-time** *minutes*]]}
	Example:
	`switch(config)# radius-server host 10.10.1.1 test username user1 password Ur2Gd2BH idle-time 3`
		Specifies parameters for individual server monitoring. The default username is test, and the default password is test. The default value for the idle timer is 0 minutes, and the valid range is from 0 to 1440 minutes.
		Note For periodic RADIUS server monitoring, you must set the idle timer to a value greater than 0.

Step 3	**radius-server deadtime** *minutes* **Example:** `switch(config)# radius-server deadtime 5`	Specifies the number of minutes before the Cisco NX-OS device checks a RADIUS server that was previously unresponsive. The default value is 0 minutes, and the valid range is from 1 to 1440 minutes.

	Command or Action	Purpose

Step 4	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 5	(Optional) **show radius-server** **Example:** `switch# show radius-server`	Displays the RADIUS server configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring RADIUS Server Hosts, on page 43

Configuring Global Periodic RADIUS Server Monitoring, on page 54

Configuring the RADIUS Dead-Time Interval

You can configure the dead-time interval for all RADIUS servers. The dead-time interval specifies the time that the Cisco NX-OS device waits after declaring a RADIUS server is dead, before sending out a test packet to determine if the server is now alive. The default value is 0 minutes.

Note When the dead-time interval is 0 minutes, RADIUS servers are not marked as dead even if they are not responding. You can configure the dead-time interval for a RADIUS server group.

Procedure

	Command or Action	Purpose

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

Step 2	**radius-server deadtime** *minutes* **Example:** `switch(config)# radius-server deadtime 5`	Configures the dead-time interval. The default value is 0 minutes. The range is from 1 to 1440 minutes.
Step 3	(Optional) **show radius** {**pending** | **pending-diff**} **Example:**	Displays the RADIUS configuration pending for distribution.

	Command or Action	Purpose
	`switch(config)# show radius pending`	
Step 4	(Optional) **radius commit** **Example:** `switch(config)# radius commit`	Applies the RADIUS configuration changes in the temporary database to the running configuration.

Step 5	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode.

Step 6	(Optional) **show radius-server** **Example:** `switch# show radius-server`	Displays the RADIUS server configuration.
Step 7	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Configuring RADIUS Server Groups, on page 47

Configuring One-Time Passwords

One-time password (OTP) support is available for Cisco NX-OS devices through the use of RSA SecurID token servers. With this feature, users authenticate to a Cisco NX-OS device by entering both a personal identification number (or one-time password) and the token code being displayed at that moment on their RSA SecurID token.

Note The token code used for logging into the Cisco NX-OS device changes every 60 seconds. To prevent problems with device discovery, we recommend using different usernames that are present on the Cisco Secure ACS internal database.

Before you begin

On the Cisco NX-OS device, configure a RADIUS server host and remote default login authentication.

Ensure that the following are installed:

- Cisco Secure Access Control Server (ACS) version 4.2
- RSA Authentication Manager version 7.1 (the RSA SecurID token server)
- RSA ACE Agent/Client

No configuration (other than a RADIUS server host and remote authentication) is required on the Cisco NX-OS device to support one-time passwords. However, you must configure the Cisco Secure ACS as follows:

1. Enable RSA SecurID token server authentication.
2. Add the RSA SecurID token server to the Unknown User Policy database.

Manually Monitoring RADIUS Servers or Groups

You can manually issue a test message to a RADIUS server or to a server group.

Procedure

	Command or Action	Purpose
Step 1	**test aaa server radius** {*ipv4-address* | *hostname*} [**vrf** *vrf-name] username password* **Example:** `switch# test aaa server radius 10.10.1.1 user1 Ur2Gd2BH`	Sends a test message to a RADIUS server to confirm availability.
Step 2	**test aaa group** *group-name username password* **Example:** `switch# test aaa group RadGroup user2 As3He3CI`	Sends a test message to a RADIUS server group to confirm availability.

Enabling or Disabling Dynamic Author Server

Procedure

	Command or Action	Purpose

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

Step 2	**aaa server radius dynamic-author** **Example:** `switch(config)# aaa server radius dynamic-author`	Enables the RADIUS dynamic author server. You can disable the RADIUS dynamic author server using the no form of this command.

Configuring RADIUS Change of Authorization

Procedure

	Command or Action	Purpose

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

Step 2	[**no**] **aaa server radius dynamic-author** **Example:** `switch(config)# aaa server radius dynamic-author`	Configures the switch as an AAA server to facilitate interaction with an external policy server. You can disable the RADIUS dynamic author and the associated clients using the no form of this command.
Step 3	[**no**] **client** {*ip-address* | **hostname** } [**server-key** [**0** | **7**] *string*] **Example:** `switch(config-locsvr-da-radius)# client 192.168.0.5 server-key cisco1`	Configures the IP address or the hostname of the AAA server client. Use the optional server-key keyword and string argument to configure the server key at the client level. You can remove the client server using the no form of this command. **Note** Configuring the server key at the client level overrides the server key that is configured at the global level.
Step 4	[**no**] **port** *port-number* **Example:** `switch(config-locsvr-da-radius)# port 3799`	Specifies the port on which a device listens to the RADIUS requests from the configured RADIUS clients. The port range is 1 - 65535. You can revert to the default port using the no form of this command. **Note** The default port for a packet of disconnect is 1700.
Step 5	[**no**] **server-key** [**0** | **7**] *string*	Configures the global RADIUS key to be shared between a device and the RADIUS clients. You can remove the server-key using the no form of this command.

Verifying the RADIUS Configuration

To display RADIUS configuration information, perform one of the following tasks:

Command	Purpose
show radius {**status** | **pending** | **pending-diff**}	Displays the RADIUS Cisco Fabric Services distribution status and other details.
show running-config radius [**all**]	Displays the RADIUS configuration in the running configuration.
show startup-config radius	Displays the RADIUS configuration in the startup configuration.
show radius-server [*hostname* | *ipv4-address*] [**directed-request** | **groups** | **sorted** | **statistics**]	Displays all configured RADIUS server parameters.

Verifying RADIUS Change of Authorization Configuration

To display RADIUS Change of Authorization configuration information, perform one of the following tasks:

Command	Purpose
show running-config dot1x	Displays the dot1x configuration in the running configuration.
show running-config aaa	Displays the AAA configuration in the running configuration.
show running-config radius	Displays the RADIUS configuration in the running configuration.
show aaa server radius statistics	Displays the local RADIUS server statistics.
show aaa client radius statistics {*ip address* | *hostname* }	Displays the local RADIUS client statistics.
clear aaa server radius statistics	Clears the local RADIUS server statistics.
clear aaa client radius statistics {*ip address* | *hostname* }	Clears the local RADIUS client statistics.

Monitoring RADIUS Servers

You can monitor the statistics that the Cisco NX-OS device maintains for RADIUS server activity.

Before you begin

Configure one or more RADIUS server hosts.

Procedure

	Command or Action	Purpose

Step 1	**show radius-server statistics** {*hostname* | *ipv4-address*}
	Example:
	`switch# show radius-server statistics 10.10.1.1`
		Displays the RADIUS statistics.

Related Topics

Configuring RADIUS Server Hosts, on page 43

Clearing RADIUS Server Statistics, on page 62

Clearing RADIUS Server Statistics

You can display the statistics that the Cisco NX-OS device maintains for RADIUS server activity.

Before you begin

Configure RADIUS servers on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose

Step 1	(Optional) **show radius-server statistics** {*hostname* | *ipv4-address*}
	Example:
	`switch# show radius-server statistics 10.10.1.1`
		Displays the RADIUS server statistics on the Cisco NX-OS device.

Step 2	**clear radius-server statistics** {*hostname* | *ipv4-address*}
	Example:
	`switch# clear radius-server statistics 10.10.1.1`
		Clears the RADIUS server statistics.

Related Topics

Configuring RADIUS Server Hosts, on page 43

Configuration Example for RADIUS

The following example shows how to configure RADIUS:

```
radius-server key 7 "ToIkLhPpG"
radius-server host 10.10.1.1 key 7 "ShMoMhTl" authentication accounting
```

```
aaa group server radius RadServer
    server 10.10.1.1
```

Configuration Examples of RADIUS Change of Authorization

The following example shows how to configure RADIUS Change of Authorization:

```
radius-server host 10.77.143.170 key 7 "fewhg123" authentication accounting
aaa server radius dynamic-author
    client 10.77.143.170 vrf management server-key 7 "fewhg123"
```

Additional References for RADIUS

This section describes additional information related to implementing RADIUS.

Related Documents

Related Topic	Document Title
Cisco NX-OS Licensing	*Cisco NX-OS Licensing Guide*
VRF configuration	*Cisco Nexus® 3550-T Unicast Routing Configuration Guide*

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
MIBs related to RADIUS	To locate and download supported MIBs, go to the following URL: ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

CHAPTER 5

Configuring IP ACLs

This chapter describes how to configure IP access control lists (ACLs) on Cisco NX-OS devices.

Unless otherwise specified, the term IP ACL refers to IPv4 ACLs.

This chapter includes the following sections:

- About ACLs, on page 65
- Prerequisites for IP ACLs, on page 68
- Guidelines and Limitations for IP ACLs, on page 69
- Default Settings for IP ACLs, on page 70
- Configuring IP ACLs, on page 70
- Verifying the IP ACL Configuration, on page 75
- Configuration Examples for IP ACLs, on page 76
- Verifying the Object-Group Configuration, on page 77
- Verifying the Time-Range Configuration, on page 77

About ACLs

An ACL is an ordered set of rules that you can use to filter traffic. Each rule specifies a set of conditions that a packet must satisfy to match the rule. When the device determines that an ACL applies to a packet, it tests the packet against the conditions of all rules. The first matching rule determines whether the packet is permitted or denied. If there is no match, the device applies the applicable implicit rule. The device continues processing packets that are permitted and drops packets that are denied.

You can use ACLs to protect networks and specific hosts from unnecessary or unwanted traffic. For example, you could use ACLs to disallow HTTP traffic from a high-security network to the Internet. You could also use ACLs to allow HTTP traffic but only to specific sites, using the IP address of the site to identify it in an IP ACL.

ACL Types and Applications

The device supports the following types of ACLs for security traffic filtering:

IPv4 ACLs

The Cisco Nexus® 3550-T device applies IPv4 ACLs only to IPv4 traffic.

IP has the following types of applications:

Router ACL
Filters Layer 3 traffic

VTY ACL
Filters virtual teletype (VTY) traffic

Note Only the ingress policy can be configured in Cisco Nexus® 3550-T switches to filter the ingress traffic based on conditions specified in the ACL on the following interfaces:

- Physical Layer 3 interfaces
- Layer 3 Ethernet port-channel interfaces
- Switch Virtual Interfaces (SVI)

This table summarizes the applications for security ACLs.

Table 8: Security ACL Applications

Application	Supported Interfaces	Types of ACLs Supported
Router ACL	• VLAN interfaces • Physical Layer 3 interfaces • Layer 3 Ethernet port-channel interfaces • Management interfaces	• IPv4 ACLs

Order of ACL Application

When the device processes a packet, it determines the forwarding path of the packet. The path determines which ACLs that the device applies to the traffic. The device only applies the Ingress router ACL.

If the packet is bridged within the ingress VLAN, the device does not apply router ACLs.

About Rules

Rules are what you create, modify, and remove when you configure how an ACL filters network traffic. Rules appear in the running configuration. When you apply an ACL to an interface or change a rule within an ACL that is already applied to an interface, the supervisor module creates ACL entries from the rules in the running configuration and sends those ACL entries to the applicable I/O module. Depending upon how you configure the ACL, there may be more ACL entries than rules, especially if you implement policy-based ACLs by using object groups when you configure rules.

You can create rules in access-list configuration mode by using the **permit** or **deny** command. The device allows traffic that matches the criteria in a permit rule and blocks traffic that matches the criteria in a deny rule. You have many options for configuring the criteria that traffic must meet in order to match the rule.

This section describes some of the options that you can use when you configure a rule.

Protocols for IP ACLs

IPv4 allows you to identify traffic by protocol. For your convenience, you can specify some protocols by name. For example, in an IPv4, you can specify ICMP by name.

You can specify any protocol by number.

In IPv4, you can specify protocols by the integer that represents the Internet protocol number.

Source and Destination

In each rule, you specify the source and the destination of the traffic that matches the rule. You can specify both the source and destination as a specific host, a network or group of hosts, or any host.

Implicit Rules for IP ACL

IP ACLs have implicit rules, which means that although these rules do not appear in the running configuration, the device applies them to traffic when no other rules in an ACL match.

All IPv4 ACLs include the following implicit rule:

```
deny ip any any
```

This implicit rule ensures that the device denies unmatched IP traffic.

This implicit rule ensures that the device denies the unmatched traffic, regardless of the protocol specified in the Layer 2 header of the traffic.

Additional Filtering Options

You can identify traffic by using additional options. These options differ by ACL type. The following list includes most but not all additional filtering options:

- IPv4 ACLs support the following additional filtering options:
  - Layer 4 protocol
  - TCP and UDP ports
  - ICMP types and codes
  - IGMP types

Sequence Numbers

The device supports sequence numbers for rules. Every rule that you enter receives a sequence number, either assigned by you or assigned automatically by the device. Sequence numbers simplify the following ACL tasks:

Adding new rules between existing rules

By specifying the sequence number, you specify where in the ACL a new rule should be positioned. For example, if you need to insert a rule between rules numbered 100 and 110, you could assign a sequence number of 105 to the new rule.

Removing a rule

Without using a sequence number, removing a rule requires that you enter the whole rule, as follows:

```
switch(config-acl)# no permit tcp 10.0.0.0/8 any
```

However, if the same rule had a sequence number of 101, removing the rule requires only the following command:

```
switch(config-acl)# no 101
```

Moving a rule

With sequence numbers, if you need to move a rule to a different position within an ACL, you can add a second instance of the rule using the sequence number that positions it correctly, and then you can remove the original instance of the rule. This action allows you to move the rule without disrupting traffic.

If you enter a rule without a sequence number, the device adds the rule to the end of the ACL and assigns a sequence number that is 10 greater than the sequence number of the preceding rule to the rule. For example, if the last rule in an ACL has a sequence number of 225 and you add a rule without a sequence number, the device assigns the sequence number 235 to the new rule.

In addition, Cisco NX-OS allows you to reassign sequence numbers to rules in an ACL. Resequencing is useful when an ACL has rules numbered contiguously, such as 100 and 101, and you need to insert one or more rules between those rules.

Logical Operators and Logical Operation Units

IP ACL rules for TCP and UDP traffic can use logical operators to filter traffic based on port numbers. Cisco NX-OS supports logical operators in only the ingress direction.

The device stores operator-operand couples in registers called logical operator units (LOUs). The LOU usage for each type of operator is as follows:

eq
Is never stored in an LOU

gt
Uses 1 LOU

lt
Uses 1 LOU

range
Uses 1 LOU

Session Manager Support for IP ACLs

Session Manager supports the configuration of IP ACLs. This feature allows you to verify ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration.

Prerequisites for IP ACLs

IP ACLs have the following prerequisites:

- You must be familiar with IP addressing and protocols to configure IP ACLs.
- You must be familiar with the interface types that you want to configure with ACLs.

Guidelines and Limitations for IP ACLs

IP ACLs have the following configuration guidelines and limitations:

- We recommend that you perform ACL configuration using the Session Manager. This feature allows you to verify ACL configuration and confirm that the resources that are required by the configuration are available before committing them to the running configuration. This recommendation is especially useful for ACLs that include more than 1000 rules.
- Duplicate ACL entries with different sequence numbers are allowed in the configuration. However, these duplicate entries are not programmed in the hardware access-list.
- Only 62 unique ACLs can be configured. Each ACL takes one label. If the same ACL is configured on multiple interfaces, the same label is shared. If each ACL has unique entries, the ACL labels are not shared, and the label limit is 62.
- Usually, ACL processing for IP packets occurs on the I/O modules, which use hardware that accelerates ACL processing. In some circumstances, processing occurs on the supervisor module, which can result in slower ACL processing, especially during processing that involves an ACL with many rules. Management interface traffic is always processed on the supervisor module. If IP packets in any of the following categories are exiting a Layer 3 interface, they are sent to the supervisor module for processing:
    - IPv4 packets that have IP options (other IP packet header fields following the destination address field).

  Rate limiters prevent redirected packets from overwhelming the supervisor module.

  .
- The VTY ACL feature restricts all traffic for all VTY lines. You cannot specify different traffic restrictions for different VTY lines. Any router ACL can be configured as a VTY ACL.
- An egress VTY ACL (an IP ACL applied to the VTY line in the outbound direction) prevents the switch from copying files using a file transfer protocol (TFTP, FTP, SCP, SFTP, etc.) unless the file transfer protocol is explicitly permitted within the egress VTY ACL.
- When you apply an undefined ACL to an interface, the system treats the ACL as empty and permits all traffic.
- ACL logging is not supported.
- The total number of IPv4 ACL flows is limited to a user-defined maximum value to prevent DoS attacks. If this limit is reached, no new logs are created until an existing flow finishes.
- A router ACL applied on a Layer 3 physical or logical interface does not match multicast traffic. If multicast traffic must be blocked, use a PACL instead.
- Only ingress RACLs are supported on Layer 3 physical interfaces and SVIs.

Default Settings for IP ACLs

This table lists the default settings for IP ACL parameters.

Table 9: Default IP ACL Parameters

Parameters	Default
IP ACLs	No IP ACLs exist by default
IP ACL entries	1024
ACL rules	Implicit rules apply to all ACLs

Configuring IP ACLs

Creating an IP ACL

You can create an IPv4 ACL on the device and add rules to it.

Before you begin

We recommend that you perform the ACL configuration using the Session Manager. This feature allows you to verify the ACL configuration and confirm that the resources that are required by the configuration are available before committing them to the running configuration. This feature is especially useful for ACLs that include more than about 1000 rules.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.
		Note When ACL is enabled only TCP and UDP packets are handled in the Cisco Nexus® 3550-T hardware.

Step 2	Enter the following commands: **ip access-list** *name*
	Example:
	`switch(config)# ip access-list acl-01`
	`switch(config-acl)#`
		Creates the IP ACL and enters IP ACL configuration mode. The *name* argument can be up to 64 characters.

Step 3	[*sequence-number*] {**permit** | **deny**} *protocol* {*source-ip-prefix* | *source-ip-mask*} {*destination-ip-prefix* | *destination-ip-mask*}	Creates a rule in the IP ACL. You can create many rules. The *sequence-number* argument

	Command or Action	Purpose
		can be a whole number between 1 and 4294967295. The **permit** and **deny** commands support many ways of identifying traffic. For IPv4 access lists, you can specify a source and destination IPv4 prefix, which matches only on the first contiguous bits, or you can specify a source and destination IPv4 wildcard mask, which matches on any bit in the address.
Step 4	(Optional) Enter the following commands: **show ip access-lists** *name* **Example:** `switch(config-acl)# show ip access-lists acl-01`	Displays the IP ACL configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config-acl)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Changing an IP ACL

You can add and remove rules in an existing IPv4 ACL, but you cannot change existing rules. Instead, to change a rule, you can remove it and recreate it with the desired changes.

If you need to add more rules between existing rules than the current sequence numbering allows, you can use the **resequence** command to reassign sequence numbers.

Before you begin

We recommend that you perform ACL configuration using the Session Manager. This feature allows you to verify ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration. This feature is especially useful for ACLs that include more than about 1000 rules.

Procedure

	Command or Action	Purpose

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

	Command or Action	Purpose

Step 2	Enter the following commands: **ip access-list** *name*
	Example:
	`switch(config)# ip access-list acl-01`
	`switch(config-acl)#`
		Enters IP ACL configuration mode for the ACL that you specify by name.

Step 3	(Optional) [*sequence-number*] {**permit** | **deny**} *protocol source destination* **Example:** `switch(config-acl)# 100 permit ip 192.168.2.0/24 any`	Creates a rule in the IP ACL. Using a sequence number allows you to specify a position for the rule in the ACL. Without a sequence number, the rule is added to the end of the rules. The *sequence-number* argument can be a whole number between 1 and 4294967295. The **permit** and **deny** commands support many ways of identifying traffic.

Step 4	(Optional) **no** {*sequence-number* | {**permit** | **deny**} *protocol source destination*}
	Example:
	`switch(config-acl)# no 80`
		Removes the rule that you specified from the IP ACL.
		The **permit** and **deny** commands support many ways of identifying traffic.

Step 5	(Optional) Enter the following commands: **show ip access-lists** *name* **Example:** `switch(config-acl)# show ip access-lists acl-01`	Displays the IP ACL configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config-acl)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Changing Sequence Numbers in an IP ACL

You can change all the sequence numbers assigned to the rules in an IP ACL.

Before you begin

We recommend that you perform ACL configuration using the Session Manager. This feature allows you to verify ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration. This feature is especially useful for ACLs that include more than about 1000 rules.

Procedure

	Command or Action	Purpose

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

Step 2	**resequence** {**ip** | **ipv4**} **access-list** *name starting-sequence-number increment* **Example:** `switch(config)# resequence access-list ip acl-01 100 10`	Assigns sequence numbers to the rules contained in the ACL, where the first rule receives the starting sequence number that you specify. Each subsequent rule receives a number larger than the preceding rule. The difference in numbers is determined by the increment that you specify. The *starting-sequence-number* argument and the *increment* argument can be a whole number between 1 and 4294967295.
Step 3	(Optional) **show ip access-lists** *name* **Example:** `switch(config)# show ip access-lists acl-01`	Displays the IP ACL configuration.
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Removing an IP ACL

You can remove an IP ACL from the device.

Before you begin

Ensure that you know whether the ACL is applied to an interface. The device allows you to remove ACLs that are currently applied. Removing an ACL does not affect the configuration of interfaces where you have applied the ACL. Instead, the device considers the removed ACL to be empty. Use the **show ip access-lists** command with the summary keyword to find the interfaces that an IP ACL is configured on.

Procedure

	Command or Action	Purpose

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

	Command or Action	Purpose
Step 2	Enter the following commands: **no ip access-list** *name* **Example:** `switch(config)# no ip access-list acl-01`	Removes the IP ACL that you specified by name from the running configuration.
Step 3	(Optional) Enter the following commands: **show ip access-lists** *name* **summary** **Example:** `switch(config)# show ip access-lists acl-01 summary`	Displays the IP ACL configuration. If the ACL remains applied to an interface, the command lists the interfaces.
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Applying an IP ACL as a Router ACL

You can apply an IPv4 ACL to any of the following types of interfaces:

- Physical Layer 3 interfaces and subinterfaces
- Layer 3 Ethernet port-channel interfaces
- VLAN interfaces
- Management interfaces

ACLs applied to these interface types are considered router ACLs.

Before you begin

Ensure that the ACL you want to apply exists and that it is configured to filter traffic in the manner that you need for this application.

Procedure

	Command or Action	Purpose

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

Step 2	Enter one of the following commands: • **interface ethernet** *slot*/*port*[**.** *number*] • **interface port-channel** *channel-number*	Enters configuration mode for the interface type that you specified.

	Command or Action	Purpose

	• **interface vlan** *vlan-id*
	• **interface mgmt** *port*
	Example:
	`switch(config)# interface ethernet 2/3`
	`switch(config-if)#`	

Step 3	Enter the following commands: **ip access-group** *access-list* {**in** | **out**} **Example:** `switch(config-if)# ip access-group acl1 in`	Applies an IPv4 ACL to the Layer 3 interface for traffic flowing in the direction specified. You can apply one router ACL per direction.
Step 4	(Optional) **show running-config aclmgr** **Example:** `switch(config-if)# show running-config aclmgr`	Displays the ACL configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Verifying the IP ACL Configuration

To display IP ACL configuration information, perform one of the following tasks.

Command	Purpose
show ip access-lists	Displays the IPv4 ACL configuration.

Command	Purpose
show running-config aclmgr [**all**]	Displays the ACL running configuration, including the IP ACL configuration and the interfaces to which IP ACLs are applied. **Note** This command displays the user-configured ACLs in the running configuration. The **all** option displays both the default (CoPP-configured) and user-configured ACLs in the running configuration.
show startup-config aclmgr [**all**]	Displays the ACL startup configuration. **Note** This command displays the user-configured ACLs in the startup configuration. The **all** option displays both the default and user-configured ACLs in the startup configuration.

Configuration Examples for IP ACLs

The following example shows how to create an IPv4 ACL named acl-01 and apply it as a port ACL to Ethernet interface 2/1, which is a Layer 2 interface:

```
ip access-list acl-01
  permit ip 192.168.2.0/24 any
interface ethernet 2/1
  ip port access-group acl-01 in
```

The following example shows how to create a VTY ACL named single-source and apply it on input IP traffic over the VTY line. This ACL allows all TCP traffic through and drops all other IP traffic:

```
ip access-list single-source
  permit tcp 192.168.7.5/24 any
  exit
  line vty
  ip access-class single-source in
```

```
show ip access-lists
```

Verifying the Object-Group Configuration

To display object-group configuration information, enter one of the following commands:

Command	Purpose
show object-group	Displays the object-group configuration.
show {**ip** } **access-lists** *name* [**expanded**]	Displays expanded statistics for the ACL configuration.
show running-config aclmgr	Displays the ACL configuration, including object groups.

Verifying the Time-Range Configuration

To display time-range configuration information, perform one of the following tasks.

Command	Purpose
show time-range	Displays the time-range configuration.
show running-config aclmgr	Displays ACL configuration, including all time ranges.

CHAPTER 6

Configuring SSH and Telnet

This chapter describes how to configure Secure Shell Protocol (SSH) and Telnet on Cisco NX-OS devices.

This chapter includes the following sections:

- About SSH and Telnet, on page 79
- Prerequisites for SSH and Telnet, on page 81
- Guidelines and Limitations for SSH and Telnet, on page 81
- Default Settings for SSH and Telnet, on page 82
- Configuring SSH , on page 82
- Configuring Telnet, on page 97
- Verifying the SSH and Telnet Configuration, on page 99
- Configuration Example for SSH, on page 99
- Configuration Example for SSH Passwordless File Copy, on page 100
- Configuration Example for X.509v3 Certificate-Based SSH Authentication, on page 102
- Additional References for SSH and Telnet, on page 103

About SSH and Telnet

This section includes information about SSH and Telnet.

SSH Server

You can use the SSH server to enable an SSH client to make a secure, encrypted connection to a Cisco NX-OS device. SSH uses strong encryption for authentication. The SSH server in the Cisco NX-OS software can interoperate with publicly and commercially available SSH clients.

The user authentication mechanisms supported for SSH are RADIUS, TACACS+, LDAP, and the use of locally stored usernames and passwords.

SSH Client

The SSH client feature is an application that runs over the SSH protocol to provide device authentication and encryption. The SSH client enables a Cisco NX-OS device to make a secure, encrypted connection to another Cisco NX-OS device or to any other device that runs the SSH server. This connection provides an outbound connection that is encrypted. With authentication and encryption, the SSH client allows for a secure communication over an insecure network.

The SSH client in the Cisco NX-OS software works with publicly and commercially available SSH servers.

SSH Server Keys

SSH requires server keys for secure communications to the Cisco NX-OS device. You can use SSH server keys for the following SSH options:

- SSH version 2 using Rivest, Shamir, and Adelman (RSA) public-key cryptography
- SSH version 2 using the Digital System Algrorithm (DSA)

Be sure to have an SSH server key-pair with the appropriate version before enabling the SSH service. You can generate the SSH server key-pair according to the SSH client version used. The SSH service accepts the following types of key-pairs for use by SSH version 2:

- The **dsa** option generates the DSA key-pair for the SSH version 2 protocol.
- The **rsa** option generates the RSA key-pair for the SSH version 2 protocol.

By default, the Cisco NX-OS software generates an RSA key using 1024 bits.

SSH supports the following public key formats:

- OpenSSH
- IETF Secure Shell (SECSH)
- Public Key Certificate in Privacy-Enhanced Mail (PEM)

⚠ **Caution** If you delete all of the SSH keys, you cannot start the SSH services.

SSH Authentication Using Digital Certificates

SSH authentication on Cisco NX-OS devices provide X.509 digital certificate support for host authentication. An X.509 digital certificate is a data item that ensures the origin and integrity of a message. It contains encryption keys for secured communications and is signed by a trusted certification authority (CA) to verify the identity of the presenter. The X.509 digital certificate support provides either DSA or RSA algorithms for authentication.

The certificate infrastructure uses the first certificate that supports the Secure Socket Layer (SSL) and is returned by the security infrastructure, either through a query or a notification. Verification of certificates is successful if the certificates are from any of the trusted CAs configured and if not revoked or expired.

You can configure your device for SSH authentication using an X.509 certificate. If the authentication fails, you are prompted for a password.

Telnet Server

The Telnet protocol enables TCP/IP connections to a host. Telnet allows a user at one site to establish a TCP connection to a login server at another site and then passes the keystrokes from one device to the other. Telnet can accept either an IP address or a domain name as the remote device address.

The Telnet server is disabled by default on the Cisco NX-OS device.

Prerequisites for SSH and Telnet

Make sure that you have configured IP on a Layer 3 interface, out-of-band on the mgmt 0 interface, or inband on an Ethernet interface.

Guidelines and Limitations for SSH and Telnet

SSH and Telnet have the following configuration guidelines and limitations:

- The Cisco NX-OS software supports only SSH version 2 (SSHv2).
- When you use the **no feature ssh feature** command, port 22 is not disabled . Port 22 is always open and a deny rule is pushed to deny all incoming external connections.
- Due to a Poodle vulnerability, SSLv3 is no longer supported.
- IPSG is not supported on the following:
    - The last six 40-Gb physical ports on the Cisco Nexus® 3550-T switches.
    - All 40G physical ports on the Cisco Nexus® 3550-T switches.
- You can configure your device for SSH authentication using an X.509 certificate. If the authentication fails, you are prompted for a password.
- The SFTP server feature does not support the regular SFTP **chown** and **chgrp** commands.
- When the SFTP server is enabled, only the admin user can use SFTP to access the device.
- SSH public and private keys imported into user accounts that are remotely authenticated through a AAA protocol (such as RADIUS or TACACS+) for the purpose of SSH Passwordless File Copy will not persist when the Nexus device is reloaded unless a local user account with the same name as the remote user account is configured on the device before the SSH keys are imported.
- SSH timeout period must be longer than the time of the tac-pac generation time. Otherwise, the VSH log might show %VSHD-2-VSHD_SYSLOG_EOL_ERR error. Ideally, set to 0 (infinity) before collecting tac-pac or showtech.

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Default Settings for SSH and Telnet

This table lists the default settings for SSH and Telnet parameters.

Table 10: Default SSH and Telnet Parameters

Parameters	Default
SSH server	Enabled
SSH server key	RSA key generated with 1024 bits
RSA key bits for generation	1024
Telnet server	Disabled
Telnet port number	23
Maximum number of SSH login attempts	3
SCP server	Disabled
SFTP server	Disabled

Configuring SSH

This section describes how to configure SSH.

Generating SSH Server Keys

You can generate an SSH server key based on your security requirements. The default SSH server key is an RSA key that is generated using 1024 bits.

Procedure

	Command or Action	Purpose

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

Step 2	**no feature ssh** **Example:** `switch(config)# no feature ssh`	Disables SSH.
Step 3	**feature ssh** **Example:**	Enables SSH.

	Command or Action	Purpose
	`switch(config)# feature ssh`	

Step 4	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 5	(Optional) **show ssh key** [**dsa** | **rsa** |] [] **Example:** `switch# show ssh key`	Displays the SSH server keys.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Specifying the SSH Public Keys for User Accounts

You can configure an SSH public key to log in using an SSH client without being prompted for a password. You can specify the SSH public key in one of these formats:

- OpenSSH format
- IETF SECSH format

Specifying the SSH Public Keys in IETF SECSH Format

You can specify the SSH public keys in IETF SECSH format for user accounts.

Before you begin

Generate an SSH public key in IETF SCHSH format.

Procedure

	Command or Action	Purpose
Step 1	**copy** *server-file* **bootflash:***filename* **Example:** `switch# copy tftp://10.10.1.1/secsh_file.pub bootflash:secsh_file.pub`	Downloads the file containing the SSH key in IETF SECSH format from a server. The server can be FTP, secure copy (SCP), secure FTP (SFTP), or TFTP.

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

	Command or Action	Purpose
Step 3	**username** *username* **sshkey file bootflash:***filename* **Example:** `switch(config)# username User1 sshkey file bootflash:secsh_file.pub`	Configures the SSH public key in IETF SECSH format.

Step 4	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 5	(Optional) **show user-account** **Example:** `switch# show user-account`	Displays the user account configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Specifying the SSH Public Keys in OpenSSH Format

You can specify the SSH public keys in OpenSSH format for user accounts.

Before you begin

Generate an SSH public key in OpenSSH format.

Procedure

	Command or Action	Purpose

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

Step 2	**username** *username* **sshkey** *ssh-key*
	Example:
	`switch(config)# username User1 sshkey ssh-rsa`
	`AAAAB3NzaC1yc2EAAAABIwAAAIEAy19oF6QaZl9G+3f1XswK3OiW4H7YyUyuA50rv7gsEPj`
	`hOBYmsi6PAVKui1nIf/DQhum+lJNqJP/eLowb7ubO+lVKRXFY/G+lJNIQW3g9igG30c6k6+`
	`XVn+NjnI1B7ihvpVh7dLddMOXwOnXHYshXmSiH3UD/vKyziEh5S4Tplx8=`
		Configures the SSH public key in OpenSSH format.

	Command or Action	Purpose

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show user-account** **Example:** `switch# show user-account`	Displays the user account configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring a Maximum Number of SSH Login Attempts

You can configure the maximum number of SSH login attempts. If the user exceeds the maximum number of permitted attempts, the session disconnects.

Note The total number of login attempts includes attempts through public-key authentication, certificate-based authentication, and password-based authentication. If public-key authentication is enabled, it takes priority. If only certificate-based and password-based authentication are enabled, certificate-based authentication takes priority. If you exceed the configured number of login attempts through all of these methods, a message appears indicating that too many authentication failures have occurred.

Procedure

	Command or Action	Purpose

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

Step 2	**ssh login-attempts** *number* **Example:** `switch(config)# ssh login-attempts 5`	Configures the maximum number of times that a user can attempt to log into an SSH session. The default maximum number of login attempts is 3. The range is from 1 to 10.

	Command or Action	Purpose
		Note The **no** form of this command removes the previous login attempts value and sets the maximum number of login attempts to the default value of 3.
Step 3	(Optional) **show running-config security all** **Example:** `switch(config)# show running-config security all`	Displays the configured maximum number of SSH login attempts.
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Starting SSH Sessions

You can start SSH sessions using IPv4 to connect to remote devices from the Cisco NX-OS device.

Before you begin

Obtain the hostname for the remote device and, if needed, the username on the remote device.

Enable the SSH server on the remote device.

Procedure

	Command or Action	Purpose
Step 1	**ssh** [*username@*]{*ipv4-address* | *hostname*} [**vrf** *vrf-name*] **Example:** `switch# ssh 10.10.1.1`	Creates an SSH IPv4 session to a remote device using IPv4.

Starting SSH Sessions from Boot Mode

You can start SSH sessions from the boot mode of the Cisco NX-OS device to connect to remote devices.

Before you begin

Obtain the hostname for the remote device and, if needed, the username on the remote device.

Enable the SSH server on the remote device.

Procedure

	Command or Action	Purpose
Step 1	**ssh** [*username***@**]*hostname* **Example:** `switch(boot)# ssh user1@10.10.1.1`	Creates an SSH session to a remote device from the boot mode of the Cisco NX-OS device.
Step 2	**exit** **Example:** `switch(boot)# exit`	Exits boot mode.
Step 3	**copy scp:**//[*username***@**]*hostname*/*filepath directory* **Example:** `switch# copy scp://user1@10.10.1.1/users abc`	Copies a file from the Cisco NX-OS device to a remote device using the Secure Copy Protocol (SCP).

Configuring SSH Passwordless File Copy

You can copy files from a Cisco NX-OS device to a secure copy (SCP) or secure FTP (SFTP) server without a password. To do so, you must create an RSA or DSA identity that consists of public and private keys for authentication with SSH.

Procedure

	Command or Action	Purpose

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

Step 2	[**no**] **username** *username* **keypair generate** {**rsa** [*bits* [**force**]] | **dsa** [**force**]}
	Example:
	`switch(config)# username user1 keypair generate rsa 2048 force`
		Generates the SSH public and private keys and stores them in the home directory ($HOME/.ssh) of the Cisco NX-OS device for the specified user. The Cisco NX-OS device uses the keys to communicate with the SSH server on the remote machine.
		The *bits* argument is the number of bits used to generate the key. The range is from 768 to 2048. The default value is 1024.
		Use the **force** keyword to replace an existing key. The SSH keys are not generated if the **force** keyword is omitted and SSH keys are already present.

Step 3	(Optional) **show username** *username* **keypair**	Displays the public key for the specified user.

	Command or Action	Purpose
	Example: `switch(config)# show username user1 keypair`	**Note** For security reasons, this command does not show the private key.
Step 4	Required: **username** *username* **keypair export** {**bootflash:***filename* | **volatile:***filename*} {**rsa** | **dsa**} [**force**] **Example:** `switch(config)# username user1 keypair export bootflash:key_rsa rsa`	Exports the public and private keys from the home directory of the Cisco NX-OS device to the specified bootflash or volatile directory. Use the **force** keyword to replace an existing key. The SSH keys are not exported if the **force** keyword is omitted and SSH keys are already present. To export the generated key pair, you are prompted to enter a passphrase that encrypts the private key. The private key is exported as the file that you specify, and the public key is exported with the same filename followed by a .pub extension. You can now copy this key pair to any Cisco NX-OS device and use SCP or SFTP to copy the public key file (*.pub) to the home directory of the server. **Note** For security reasons, this command can be executed only from global configuration mode.
Step 5	Required: **username** *username* **keypair import** {**bootflash:***filename* | **volatile:***filename*} {**rsa** | **dsa**} [**force**] **Example:** `switch(config)# username user1 keypair import bootflash:key_rsa rsa`	Imports the exported public and private keys from the specified bootflash or volatile directory to the home directory of the Cisco NX-OS device. Use the **force** keyword to replace an existing key. The SSH keys are not imported if the **force** keyword is omitted and SSH keys are already present. To import the generated key pair, you are prompted to enter a passphrase that decrypts the private key. The private key is imported as the file that you specify, and the public key is imported with the same filename followed by a .pub extension. **Note** For security reasons, this command can be executed only from global configuration mode. **Note** Only the users whose keys are configured on the server are able to access the server without a password.

What to do next

On the SCP or SFTP server, use the following command to append the public key stored in the *.pub file (for example, key_rsa.pub) to the authorized_keys file:

$ cat key_rsa.pub >> $HOME/.ssh/ authorized_keys

You can now copy files from the Cisco NX-OS device to the server without a password using standard SSH and SCP commands.

Configuring SCP and SFTP Servers

You can configure an SCP or SFTP server on the Cisco NX-OS device in order to copy files to and from a remote device. After you enable the SCP or SFTP server, you can execute an SCP or SFTP command on the remote device to copy the files to or from the Cisco NX-OS device.

Note The arcfour and blowfish cipher options are not supported for the SCP server.

Procedure

	Command or Action	Purpose

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

Step 2	[**no**] **feature scp-server** **Example:** `switch(config)# feature scp-server`	Enables or disables the SCP server on the Cisco NX-OS device.
Step 3	Required: [**no**] **feature sftp-server** **Example:** `switch(config)# feature sftp-server`	Enables or disables the SFTP server on the Cisco NX-OS device.

Step 4	Required: **exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 5	(Optional) **show running-config security** **Example:** `switch# show running-config security`	Displays the configuration status of the SCP and SFTP servers.

Step 6	(Optional) **copy running-config startup-config**
	Example:
	`switch# copy running-config`
	`startup-config`
		Copies the running configuration to the startup configuration.

Configuring X.509v3 Certificate-Based SSH Authentication

You can configure SSH authentication using X.509v3 certificates.

Before you begin

Enable the SSH server on the remote device.

Procedure

	Command or Action	Purpose

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

Step 2	**username** *user-id* [**password** [**0** | **5**] *password*] **Example:** `switch(config)# username jsmith password 4Ty18Rnt`	Configures a user account. The *user-id* argument is a case-sensitive, alphanumeric character string with a maximum length of 28 characters. Valid characters are uppercase letters A through Z, lowercase letters a through z, numbers 0 through 9, hyphen (-), period (.), underscore (_), plus sign (+), and equal sign (=). The at symbol (@) is supported in remote usernames but not in local usernames. Usernames must begin with an alphanumeric character. The default password is undefined. The **0** option indicates that the password is clear text, and the **5** option indicates that the password is encrypted. The default is **0** (clear text). **Note** If you do not specify a password, the user might not be able to log in to the Cisco NX-OS device. **Note** If you create a user account with the encrypted password option, the corresponding SNMP user will not be created.

Step 3	**username** *user-id* **ssh-cert-dn** *dn-name* {**dsa** | **rsa**}
	Example:
	`switch(config)# username jsmith ssh-cert-dn "/O = ABCcompany, OU = ABC1, emailAddress = jsmith@ABCcompany.com, L = Metropolis, ST = New York, C = US, CN = jsmith" rsa`
		Specifies an SSH X.509 certificate distinguished name and DSA or RSA algorithm to use for authentication for an existing user account. The distinguished name can be up to 512 characters and must follow the format shown in the examples. Make sure the email address and state are configured as emailAddress and ST, respectively.

Step 4	[**no**] **crypto ca trustpoint** *trustpoint*	Configures a trustpoint.

	Command or Action	Purpose

	Example:
	`switch(config)# crypto ca trustpoint winca`
	`switch(config-trustpoint)#`
		Note Before you delete a trustpoint using the **no** form of this command, you must first delete the CRL and CA certificate, using the **delete crl** and **delete ca-certificate** commands.

Step 5	**crypto ca authenticate** *trustpoint* **Example:** `switch(config-trustpoint)# crypto ca authenticate winca`	Configures a CA certificate for the trustpoint. **Note** To delete a CA certificate, enter the **delete ca-certificate** command in the trustpoint configuration mode.
Step 6	(Optional) **crypto ca crl request** *trustpoint* **bootflash:***static-crl***.crl** **Example:** `switch(config-trustpoint)# crypto ca crl request winca bootflash:crllist.crl`	This command is optional but highly recommended. Configures the certificate revocation list (CRL) for the trustpoint. The CRL file is a snapshot of the list of revoked certificates by the trustpoint. This static CRL list is manually copied to the device from the Certification Authority (CA). **Note** Static CRL is the only supported revocation check method. **Note** To delete the CRL, enter the **delete crl** command.
Step 7	(Optional) **show crypto ca certificates** **Example:** `switch(config-trustpoint)# show crypto ca certificates`	Displays the configured certificate chain and associated trustpoint.
Step 8	(Optional) **show crypto ca crl** *trustpoint* **Example:** `switch(config-trustpoint)# show crypto ca crl winca`	Displays the contents of the CRL list of the specified trustpoint.
Step 9	(Optional) **show user-account** **Example:** `switch(config-trustpoint)# show user-account`	Displays configured user account details.
Step 10	(Optional) **show users** **Example:** `switch(config-trustpoint)# show users`	Displays the users logged into the device.

	Command or Action	Purpose
Step 11	(Optional) **copy running-config startup-config** **Example:** `switch(config-trustpoint)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring Legacy SSH Algorithm Support

You can configure support for legacy SSH security algorithms, message authentication codes (MACs), key types, and ciphers.

Procedure

	Command or Action	Purpose

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

Step 2	(Optional) **ssh kexalgos** [**all**] **Example:** `switch(config)# ssh kexalgos all`	Use the **all** keyword to enable all supported KexAlgorithms which are the key exchange methods that are used to generate per-connection keys. Supported KexAlgorithmns are: • curve25519-sha256 • diffie-hellman-group-exchange-sha256 • diffie-hellman-group1-sha1 • diffie-hellman-group14-sha1 • diffie-hellman-group1-sha1 • ecdh-sha2-nistp256 • ecdh-sha2-nistp384
Step 3	(Optional) **ssh macs all** **Example:** `switch(config)# ssh macs all`	Enables all supported MACs which are the message authentication codes used to detect traffic modification. Supported MACs are: • hmac-sha1
Step 4	(Optional) **ssh ciphers** [**all**] **Example:**	Use the **all** keyword to enable all supported ciphers to encrypt the connection.

	Command or Action	Purpose
	`switch(config)# ssh ciphers all`	Supported ciphers are: • aes128-cbc • aes192-cbc • aes256-cbc • aes128-ctr • aes192-ctr • aes256-ctr • aes256-gcm@openssh.com • aes128-gcm@openssh.com
Step 5	(Optional) **ssh keytypes all** **Example:** `switch(config)# ssh keytypes all`	Enables all supported PubkeyAcceptedKeyTypes which are the public key algorithms that the server can use to authenticate itself to the client. Supported key types are: • ssh-dss • ssh-rsa

Algorithms Supported - FIPs Mode Enabled

The list of algorithms supported when the FIPs mode is enabled are as follows:

Table 11: Algorithms Supported - FIPs Mode Enabled

Algorithms	Supported	Unsupported
ciphers	• aes128-ctr • aes256-ctr • aes256-gcm@openssh.com • aes128-gcm@openssh.com	• aes192-ctr • aes128-cbc • aes192-cbc • aes256-cbc
hmac	• hmac-sha2-256 • hmac-sha2-512 • hmac-sha1	• hmac-sha2-256-etm@openssh.com • hmac-sha2-512-etm@openssh.com • hmac-sha1-etm@openssh.com

Algorithms	Supported	Unsupported
kexalgo	• ecdh-sha2-nistp256 • ecdh-sha2-nistp384 • ecdh-sha2-nistp521 • diffie-hellman-group16-sha512 • diffie-hellman-group14-sha1 • diffie-hellman-group14-sha256	• curve25519-sha256 • curve25519-sha256@libssh.org
keytypes	• rsa-sha2-256 • ecdsa-sha2-nistp256 • ecdsa-sha2-nistp384 • ecdsa-sha2-nistp521	ssh-rsa

Changing the Default SSH Server Port

You can change the SSHv2 port number from the default port number 22. Encryptions used while changing the default SSH port provides you with connections that support stronger privacy and session integrity

Procedure

	Command or Action	Purpose

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

Step 2	**no feature ssh** **Example:** `switch(config)# no feature ssh`	Disables SSH.

Step 3	**show sockets** *local-port-range*
	Example:
	`switch(config)# show sockets local port range (15001 - 58000)`
	`switch(config)# local port range (58001 - 63535) and nat port range (63536 - 65535)`
		Displays the available port range.

Step 4	**ssh port** *local-port* **Example:** `switch(config)# ssh port 58003`	Configures the port.

	Command or Action	Purpose
Step 5	**feature ssh** **Example:** `switch(config)# feature ssh`	Enables SSH.

Step 6	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 7	(Optional) **show running-config security all** **Example:** `switch# ssh port 58003`	Displays the security configuration.
Step 8	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Clearing SSH Hosts

When you download a file from a server using SCP or SFTP, or when you start an SSH session from this device to a remote host, you establish a trusted SSH relationship with that server. You can clear the list of trusted SSH servers for your user account.

Procedure

	Command or Action	Purpose
Step 1	**clear ssh hosts** **Example:** `switch# clear ssh hosts`	Clears the SSH host sessions and the known host file.

Disabling the SSH Server

By default, the SSH server is enabled on the Cisco NX-OS device. You can disable the SSH server to prevent SSH access to the switch.

Procedure

	Command or Action	Purpose

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

	Command or Action	Purpose
Step 2	**no feature ssh** **Example:** `switch(config)# no feature ssh`	Disables SSH.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show ssh server** **Example:** `switch# show ssh server`	Displays the SSH server configuration.

Step 5	(Optional) **copy running-config startup-config**
	Example:
	`switch# copy running-config`
	`startup-config`
		Copies the running configuration to the startup configuration.

Deleting SSH Server Keys

You can delete SSH server keys on the Cisco NX-OS device after you disable the SSH server.

Note To reenable SSH, you must first generate an SSH server key.

Procedure

	Command or Action	Purpose

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

Step 2	**no feature ssh** **Example:** `switch(config)# no feature ssh`	Disables SSH.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

	Command or Action	Purpose
Step 4	(Optional) **show ssh key** **Example:** `switch# show ssh key`	Displays the SSH server key configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

Generating SSH Server Keys, on page 82

Clearing SSH Sessions

You can clear SSH sessions from the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	**show users** **Example:** `switch# show users`	Displays user session information.
Step 2	**clear line** *vty-line* **Example:** `switch(config)# clear line pts/12`	Clears a user SSH session.

Configuring Telnet

This section describes how to configure Telnet on the Cisco NX-OS device.

Enabling the Telnet Server

You can enable the Telnet server on the Cisco NX-OS device. By default, the Telnet server is disabled.

Procedure

	Command or Action	Purpose

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

	Command or Action	Purpose
Step 2	**feature telnet** **Example:** `switch(config)# feature telnet`	Enables the Telnet server. The default is disabled.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show telnet server** **Example:** `switch# show telnet server`	Displays the Telnet server configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Starting Telnet Sessions to Remote Devices

You can start Telnet sessions to connect to remote devices from the Cisco NX-OS device. You can start Telnet sessions using either IPv4.

Before you begin

Obtain the hostname or IP address for the remote device and, if needed, the username on the remote device.

Enable the Telnet server on the Cisco NX-OS device.

Enable the Telnet server on the remote device.

Procedure

	Command or Action	Purpose
Step 1	**telnet** {*ipv4-address* | *host-name*} [*port-number*] **Example:** `switch# telnet 10.10.1.1`	Starts a Telnet session to a remote device using IPv4. The default port number is 23. The range is from 1 to 65535.

Related Topics

Enabling the Telnet Server, on page 97

Clearing Telnet Sessions

You can clear Telnet sessions from the Cisco NX-OS device.

Before you begin

Enable the Telnet server on the Cisco NX-OS device.

Procedure

	Command or Action	Purpose
Step 1	**show users** **Example:** `switch# show users`	Displays user session information.
Step 2	**clear line** *vty-line* **Example:** `switch(config)# clear line pts/12`	Clears a user Telnet session.

Verifying the SSH and Telnet Configuration

To display the SSH and Telnet configuration information, perform one of the following tasks:

Command	Purpose
show ssh key [**dsa** | **rsa**] []	Displays the SSH server keys.
show running-config security [**all**]	Displays the SSH and user account configuration in the running configuration. The **all** keyword displays the default values for the SSH and user accounts.
show ssh server	Displays the SSH server configuration.
show telnet server	Displays the Telnet server configuration.
show username *username* **keypair**	Displays the public key for the specified user.
show user-account	Displays configured user account details.
show users	Displays the users logged into the device.

Configuration Example for SSH

The following example shows how to configure SSH with an OpenSSH key:

Procedure

Step 1 Disable the SSH server.

Example:

```
switch# configure terminal
switch(config)# no feature ssh
```

Step 2 Generate an SSH server key.

Example:

```
switch(config)# ssh key rsa
generating rsa key(1024 bits)......
generated rsa key
```

Step 3 Enable the SSH server.

Example:

```
switch(config)# feature ssh
```

Step 4 Display the SSH server key.

Example:

Step 5 Specify the SSH public key in OpenSSH format.

Example:

```
switch(config)# username User1 sshkey ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAy19oF6QaZl9G+3f1XswK3OiW4H7YyUyuA50r
v7gsEPjhOBYmsi6PAVKui1nIf/DQhum+lJNqJP/eLowb7ubO+lVKRXFY/G+lJNIQ
W3g9igG30c6k6+XVn+NjnI1B7ihvpVh7dLddMOXwOnXHYshXmSiH3UD/vKyziEh5
4Tplx8=
```

Step 6 Save the configuration.

Example:

```
switch(config)# copy running-config startup-config
```

Configuration Example for SSH Passwordless File Copy

The following example shows how to copy files from a Cisco NX-OS device to a secure copy (SCP) or secure FTP (SFTP) server without a password:

Procedure

Step 1 Generate the SSH public and private keys and store them in the home directory of the Cisco NX-OS device for the specified user.

Example:

```
switch# configure terminal
switch(config)# username admin keypair generate rsa
generating rsa key(1024 bits)......
```

```
generated rsa key
```

Step 2 Display the public key for the specified user.

Example:

```
switch(config)# show username admin keypair

**************************************

rsa Keys generated: Thu Jul  9 11:10:29 2013

ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAxWmjJT+oQhIcvnrMbx2BmD0P8boZElTfJ
Fx9fexWp6rOiztlwODtehnjadWc6A+DE2DvYNvqsrU9TBypYDPQkR/+Y6cKubyFW
VxSBG/NHztQc3+QC1zdkIxGNJbEHyFoajzNEO8LLOVFIMCZ2Td7gxUGRZc+fbq
S33GZsCAX6v0=

bitcount:262144
fingerprint:
8d:44:ee:6c:ca:0b:44:95:36:d0:7d:f2:b5:78:74:7d
**************************************

could not retrieve dsa key information
**************************************
```

Step 3 Export the public and private keys from the home directory of the Cisco NX-OS device to the specified bootflash directory.

Example:

```
switch(config)# username admin keypair export bootflash:key_rsa rsa
Enter Passphrase:
switch(config)# dir
.
.
.
        951     Jul 09 11:13:59 2013  key_rsa
        221     Jul 09 11:14:00 2013  key_rsa.pub
.
.
```

Step 4 After copying these two files to another Cisco NX-OS device using the **copy scp** or **copy sftp** command, import them to the home directory of the Cisco NX-OS device.

Example:

```
switch(config)# username admin keypair import bootflash:key_rsa rsa
Enter Passphrase:
switch(config)# show username admin keypair
**************************************

rsa Keys generated: Thu Jul  9 11:10:29 2013

ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAxWmjJT+oQhIcvnrMbx2BmD0P8boZElTfJ
Fx9fexWp6rOiztlwODtehnjadWc6A+DE2DvYNvqsrU9TBypYDPQkR/+Y6cKubyFW
VxSBG/NHztQc3+QC1zdkIxGNJbEHyFoajzNEO8LLOVFIMCZ2Td7gxUGRZc+fbq
S33GZsCAX6v0=
bitcount:262144
fingerprint:
8d:44:ee:6c:ca:0b:44:95:36:d0:7d:f2:b5:78:74:7d
**************************************

could not retrieve dsa key information
**************************************
switch(config)#
```

Step 5 On the SCP or SFTP server, append the public key stored in key_rsa.pub to the authorized_keys file.

Example:

```
$ cat key_rsa.pub >> $HOME/.ssh/ authorized_keys
```

You can now copy files from the Cisco NX-OS device to the server without a password using standard SSH and SCP commands.

Step 6 (Optional) Repeat this procedure for the DSA keys.

Configuration Example for X.509v3 Certificate-Based SSH Authentication

The following example shows how to configure SSH authentication using X.509v3 certificates:

```
configure terminal
username jsmith password 4Ty18Rnt
username jsmith ssh-cert-dn "/O = ABCcompany, OU = ABC1,
emailAddress = jsmith@ABCcompany.com, L = Metropolis, ST = New York, C = US, CN = jsmith"
rsa
crypto ca trustpoint tp1
crypto ca authenticate tp1
crypto ca crl request tp1 bootflash:crl1.crl

show crypto ca certificates
Trustpoint: tp1
CA certificate 0:
subject= /CN=SecDevCA
issuer= /CN=SecDevCA
serial=01AB02CD03EF04GH05IJ06KL07MN
notBefore=Jun 29 12:36:26 2016 GMT
notAfter=Jun 29 12:46:23 2021 GMT
SHA1 Fingerprint=47:29:E3:00:C1:C1:47:F2:56:8B:AC:B2:1C:64:48:FC:F4:8D:53:AF
purposes: sslserver sslclient

show crypto ca crl tp1
Trustpoint: tp1 CRL: Certificate Revocation List (CRL):
    Version 2 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
    Issuer: /CN=SecDevCA
    Last Update: Aug 8 20:03:15 2016 GMT
    Next Update: Aug 16 08:23:15 2016 GMT
    CRL extensions:
        X509v3 Authority Key Identifier:
            keyid:30:43:AA:80:10:FE:72:00:DE:2F:A2:17:E4:61:61:44:CE:78:FF:2A
```

```
show user-account
user:user1
        this user account has no expiry date
        roles:network-operator
        ssh cert DN : /C = US, ST = New York, L = Metropolis, O = cisco , OU = csg, CN =
user1; Algo: x509v3-sign-rsa

show users
NAME      LINE        TIME          IDLE      PID        COMMENT
user1     pts/1       Jul 27 18:43  00:03     18796      (10.10.10.1)   session=ssh
```

Additional References for SSH and Telnet

This section describes additional information related to implementing SSH and Telnet.

Related Documents

Related Topic	Document Title
Cisco NX-OS licensing	*Cisco NX-OS Licensing Guide*
VRF configuration	*Cisco Nexus® 3550-T Unicast Routing Configuration Guide*

MIBs

MIBs	MIBs Link
MIBs related to SSH and Telnet	To locate and download supported MIBs, go to the following URL: ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

CHAPTER 7

Configuring DHCP

This chapter describes how to configure the Dynamic Host Configuration Protocol (DHCP) on a Cisco NX-OS device.

This chapter includes the following sections:

- About DHCP Client, on page 105
- Guidelines and Limitations for DHCP, on page 105
- Enabling DHCP Client, on page 105
- Configuration Examples for DHCP Client, on page 106

About DHCP Client

The DHCP client feature enables the configuration of an IPv4 address on the management port.

Guidelines and Limitations for DHCP

DHCP has the following configuration guidelines and limitations:

- Only DHCP client is supported.
- No support for DHCPv6 (or IPv6).
- PowerOn Auto Provisioning (POAP) can be used for the DHCP client. Restrictions for POAP:
    - POAP is supported only on the management port.
    - No support for IPv6.

  For more details about POAP, see the Fundamentals Guide.

Enabling DHCP Client

You can use the DHCP client feature to enable the configuration of an IPv4 address on an interface.

Note DHCP client is independent of the DHCP relay processes, so it does not require that the **feature dhcp** command be enabled.

Procedure

	Command or Action	Purpose

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

Step 2	**interface mgmt 0**
	Example:
	`switch(config)# interface mgmt 0`
	`switch(config-if)#`
		• Enters interface configuration mode and specifies the management interface as the interface for which you want to enable the DHCP client feature.

Step 3	[**no**] {**ip** } **address dhcp** **Example:** `switch(config-if)# ip address dhcp`	Assigns an IPv4 address to the interface. The **no** form of this command releases the IP address.
Step 4	(Optional) Run the **show running-config interface mgmt 0** command.	Displays the IPv4 address assigned to the interface in the running configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	Copies the running configuration to the startup configuration. Only the {**ip**} **address dhcp** command is saved. The assigned IP address is not saved even though it shows in the running configuration.

Configuration Examples for DHCP Client

The following example shows how to use the DHCP client feature:

```
switch# configure terminal
switch(config)# interface mgmt 0
switch(config-if)# no shutdown
switch(config-if)# ip address dhcp
switch(config-if)# show running-config interface vlan 7
```