Configuring SSH and Telnet

This chapter contains the following sections:

Information About SSH and Telnet

SSH Server

The Secure Shell Protocol (SSH) server feature enables a SSH client to make a secure, encrypted connection to a Cisco Nexus device. SSH uses strong encryption for authentication. The SSH server in the Cisco Nexus device switch interoperates with publicly and commercially available SSH clients.

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

SSH Client

The SSH client feature is an application running over the SSH protocol to provide device authentication and encryption. The SSH client enables a switch to make a secure, encrypted connection to another Cisco Nexus device or to any other device running an 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 Nexus device 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)
- SSH version 2 using the Elliptic Curve Digital Signature Algorithm (ECDSA)

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.
- The **ecdsa** option generates the ECDSA 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.

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

You can configure SSH authentication using X.509v3 certificates (RFC 6187). X.509v3 certificate-based SSH authentication uses certificates combined with a smartcard to enable two-factor authentication for Cisco device access. The SSH client is provided by Cisco partner Pragma Systems.

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 system to the other. Telnet can accept either an IP address or a domain name as the remote system address.

The Telnet server is enabled by default on the Cisco Nexus device.

Guidelines and Limitations for SSH

SSH has the following configuration guidelines and limitations:

- The Cisco Nexus device supports only SSH version 2 (SSHv2).
- 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.

Configuring 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	switch# **configure terminal**	Enters global configuration mode.
Step 2	**ssh key** {**dsa** [**force**] | **rsa** [*bits*[**force**]] | **ecdsa** [*bits* [**force**]]}	Generates the SSH server key. The *bits* argument is the number of bits used to generate the RSA key. The range is from 768 to 2048. The default value is 1024. You cannot specify the size of the DSA key. It is always set to 1024 bits. Use the **force** keyword to replace an existing key. **Note** If you configure ssh key dsa, you must do the following additional configurations: ssh keytypes all and ssh kexalgos all
Step 3	**ssh rekey max-data** *max-data* **max-time** *max-time*i	Configures the rekey parameters.

	Command or Action	Purpose
	Example: `switch(config)# ssh rekey max-data 1K max-time 1M`	
Step 4	**feature ssh** **Example:** `switch(config)# feature ssh`	Enables SSH.
Step 5	switch(config)# **exit**	Exits global configuration mode.
Step 6	(Optional) **show ssh key** [**dsa** | **rsa** | **ecdsa**] [**md5**] **Example:** `switch# show ssh key`	Displays the SSH server keys. This command displays the fingerprint in SHA256 format by default. SHA256 is more secure than the old default format of MD5. However, the **md5** option has been added, if you want to see the fingerprint in MD5 format for backward compatibility.
Step 7	**show run security all**	
Step 8	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

The following example shows how to generate an SSH server key:

```
switch# configure terminal
switch(config)# ssh key rsa 2048
switch(config)# exit
switch# show ssh key
switch# copy running-config startup-config
```

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 three different formats:

- Open SSH format
- IETF SECSH format
- Public Key Certificate in PEM format

Specifying the SSH Public Keys in Open SSH Format

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **username** *username* **sshkey** *ssh-key*	Configures the SSH public key in SSH format.
Step 3	switch(config)# **exit**	Exits global configuration mode.
Step 4	(Optional) switch# **show user-account**	Displays the user account configuration.
Step 5	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

The following example shows how to specify an SSH public key in open SSH format:

```
switch# configure terminal
switch(config)# username User1 sshkey ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAri3mQy4W1AV9Y2t2hrEWgbUEYz
CfTPO5B8LRkedn56BEy2N9ZcdpqE6aqJLZwfZcTFEzaAAZp9AS86dgBAjsKGs7UxnhGySr8ZELv+DQBsDQH6rZt0KR+2Da8hJD4Z
XIeccWk0gS1DQUNZ300xstQsYZUtqnx1bvm5Ninn0McNinn0Mc=
switch(config)# exit
switch# show user-account
switch# copy running-config startup-config
```

Note The **username** command in the example above is a single line that has been broken for legibility.

Specifying the SSH Public Keys in IETF SECSH Format

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

Procedure

	Command or Action	Purpose
Step 1	switch# **copy** *server-file* **bootflash:** *filename*	Downloads the file that contains the SSH key in IETF SECSH format from a server. The server can be FTP, SCP, SFTP, or TFTP.
Step 2	switch# **configure terminal**	Enters global configuration mode.
Step 3	switch(config)# **username** *username* **sshkey file** *filename*	Configures the SSH public key in SSH format.
Step 4	switch(config)# **exit**	Exits global configuration mode.
Step 5	(Optional) switch# **show user-account**	Displays the user account configuration.

	Command or Action	Purpose
Step 6	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

The following example shows how to specify the SSH public key in the IETF SECSH format:

```
switch#copy tftp://10.10.1.1/secsh_file.pub bootflash:secsh_file.pub
switch# configure terminal
switch(config)# username User1 sshkey file bootflash:secsh_file.pub
switch(config)# exit
switch# show user-account
switch# copy running-config startup-config
```

Specifying the SSH Public Keys in PEM-Formatted Public Key Certificate Form

You can specify the SSH public keys in PEM-formatted Public Key Certificate form for user accounts.

Procedure

	Command or Action	Purpose
Step 1	switch# **copy** *server-file* **bootflash:** *filename*	Downloads the file that contains the SSH key in PEM-formatted Public Key Certificate form from a server. The server can be FTP, SCP, SFTP, or TFTP
Step 2	switch# **configure terminal**	Enters global configuration mode.
Step 3	(Optional) switch# **show user-account**	Displays the user account configuration.
Step 4	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

The following example shows how to specify the SSH public keys in PEM-formatted public key certificate form:

```
switch# copy tftp://10.10.1.1/cert.pem bootflash:cert.pem
switch# configure terminal
switch# show user-account
switch# copy running-config startup-config
```

Configuring the SSH Source Interface

You can configure SSH to use a specific interface.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **ip ssh source-interface** *type slot/port*	Configures the source interface for all SSH packets. The following list contains the valid values for *interface*. • ethernet • loopback • mgmt • port-channel • vlan
Step 3	switch(config)# **show ip ssh source-interface**	Displays the configured SSH source interface.

Example

This example shows how to configure the SSH source interface:

```
switch(config)# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip ssh source-interface ethernet 1/7
switch(config)# show ip ssh source-interface
VRF Name                                Interface
default                                 Ethernet1/7
```

Starting SSH Sessions to Remote Devices

You can start SSH sessions to connect to remote devices from your Cisco Nexus device.

Procedure

	Command or Action	Purpose
Step 1	switch# **ssh** {*hostname* | *username@hostname*} [**vrf** *vrf-name*]	Creates an SSH session to a remote device. The *hostname* argument can be an IPv4 address or a hostname.

Clearing SSH Hosts

When you download a file from a server using SCP or SFTP, you establish a trusted SSH relationship with that server.

Procedure

	Command or Action	Purpose
Step 1	switch# **clear ssh hosts**	Clears the SSH host sessions.

Disabling the SSH Server

By default, the SSH server is enabled on the Cisco Nexus device.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **[no] feature ssh**	Enables/disables the SSH server. The default is enabled.
Step 3	switch(config)# **exit**	Exits global configuration mode.
Step 4	(Optional) switch# **show ssh server**	Displays the SSH server configuration.
Step 5	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Deleting SSH Server Keys

You can delete SSH server keys 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	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **no feature ssh**	Disables the SSH server.
Step 3	switch(config)# **no ssh key** [**dsa** | **rsa**]	Deletes the SSH server key. The default is to delete all the SSH keys.
Step 4	switch(config)# **exit**	Exits global configuration mode.
Step 5	(Optional) switch# **show ssh key**	Displays the SSH server configuration.
Step 6	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Clearing SSH Sessions

You can clear SSH sessions from the Cisco Nexus device.

Procedure

	Command or Action	Purpose
Step 1	switch# **show users**	Displays user session information.
Step 2	switch# **clear line** *vty-line*	Clears a user SSH session.

Configuration Examples for SSH

The following example shows how to configure SSH:

Procedure

Step 1 Generate an SSH server key.

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

Step 2 Enable the SSH server.

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

Note This step should not be required because the SSH server is enabled by default.

Step 3 Display the SSH server key.

```
switch(config)# show ssh key
rsa Keys generated:Fri May  8 22:09:47 2009

ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAri3mQy4W1AV9Y2t2hrEWgbUEYzCfTPO5B8LRkedn56BEy2N9ZcdpqE6aqJLZwfZ/
cTFEzaAAZp9AS86dgBAjsKGs7UxnhGySr8ZELv+DQBsDQH6rZt0KR+2Da8hJD4ZXIeccWk0gS1DQUNZ300xstQsYZUtqnx1bvm5/
Ninn0Mc=

bitcount:1024
fingerprint:
4b:4d:f6:b9:42:e9:d9:71:3c:bd:09:94:4a:93:ac:ca
***********************************
could not retrieve dsa key information
***********************************
```

Step 4 Specify the SSH public key in Open SSH format.

```
switch(config)# username User1 sshkey ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAri3mQy4W1AV9Y2t2hrEWgbUEYz

CfTPO5B8LRkedn56BEy2N9ZcdpqE6aqJLZwfZcTFEzaAAZp9AS86dgBAjsKGs7UxnhGySr8ZELv+DQBsDQH6rZt0KR+2Da8hJD4Z

XIeccWk0gS1DQUNZ300xstQsYZUtqnx1bvm5Ninn0McNinn0Mc=
```

Step 5 Save the configuration.

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

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.

	Command or Action	Purpose
		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* **Example:** `switch(config)# crypto ca trustpoint winca`	Configures a trustpoint.
Step 5	[**no**] **crypto ca authentication** *trustpoint* **Example:** `switch(config)# crypto ca authentication winca`	Configures a certificate chain for the trustpoint.
Step 6	**crypto ca crl request** *trustpoint* **bootflash:***static-crl***.crl** **Example:** `switch(config)# crypto ca crl request winca bootflash:crllist.crl`	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.
Step 7	(Optional) **show crypto ca certificates** **Example:** `switch(config)# show crypto ca certificates`	Displays the configured certificate chain and associated trustpoint.
Step 8	(Optional) **show crypto ca crl** *trustpoint* **Example:** `switch(config)# 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)# show user-account`	Displays configured user account details.

	Command or Action	Purpose
Step 10	(Optional) **show users** **Example:** `switch(config)# show users`	Displays the users logged into the device.
Step 11	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

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 authentication 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
```

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 global configuration mode.

Step 2	(Optional) **ssh kexalgos all** **Example:** `switch(config)# ssh kexalgos all`	Enables 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-group14-sha1 • diffie-hellman-group1-sha1 • ecdh-sha2-nistp256 • ecdh-sha2-nistp384 • ecdh-sha2-nistp521
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 • hmac-sha2-256 • hmac-sha2-512 • aes256-gcm@openssh.com • aes128-gcm@openssh.com

	Command or Action	Purpose
Step 4	(Optional) **ssh ciphers all** **Example:** `switch(config)# ssh ciphers all`	Enables all supported ciphers to encrypt the connection. 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: • ecdsa-sha2-nistp256 • ecdsa-sha2-nistp384 • ecdsa-sha2-nistp521 • ssh-dss • ssh-rsa

Changing the Default SSH Server Port

Beginning with Cisco NX-OS Cisco Release 9.2(1), 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.

	Command or Action	Purpose
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.
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.

Configuring Telnet

Enabling the Telnet Server

By default, the Telnet server is enabled. You can disable the Telnet server on your Cisco Nexus device.

Procedure

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

	Command or Action	Purpose
Step 2	switch(config)# **[no] feature telnet**	Enables/disables the Telnet server. The default is enabled.

Reenabling the Telnet Server

If the Telnet server on your Cisco Nexus device has been disabled, you can reenable it.

Procedure

	Command or Action	Purpose
Step 1	switch(config)# **[no] feature telnet**	Reenables the Telnet server.

Configuring the Telnet Source Interface

You can configure Telnet to use a specific interface.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **ip telnet source-interface** *type slot/port*	Configures the source interface for all Telnet packets. The following list contains the valid values for *interface*. • ethernet • loopback • mgmt • port-channel • vlan

Example

This example shows how to configure the Telnet source interface:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip telnet source-interface ethernet 1/6
switch(config)# show ip telnet source-interface
VRF Name                                Interface
default                                 Ethernet1/6
switch(config)#
```

Starting Telnet Sessions to Remote Devices

Before you start a Telnet session to connect to remote devices, you should do the following:

- Obtain the hostname for the remote device and, if needed, obtain the username on the remote device.
- Enable the Telnet server on the Cisco Nexus device.
- Enable the Telnet server on the remote device.

Procedure

	Command or Action	Purpose
Step 1	switch# **telnet** *hostname*	Creates a Telnet session to a remote device. The *hostname* argument can be an IPv4 address, an IPv6 address, or a device name.

Example

The following example shows how to start a Telnet session to connect to a remote device:

```
switch# telnet 10.10.1.1
Trying 10.10.1.1...
Connected to 10.10.1.1.
Escape character is '^]'.
switch login:
```

Clearing Telnet Sessions

You can clear Telnet sessions from the Cisco Nexus device.

Procedure

	Command or Action	Purpose
Step 1	switch# **show users**	Displays user session information.
Step 2	switch# **clear line** *vty-line*	Clears a user Telnet session.

Verifying the SSH and Telnet Configuration

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

Command or Action	Purpose
switch# **show ssh key [dsa | rsa][md5]**	Displays SSH server keys. This command displays the fingerprint in SHA256 format by default. SHA256 is more secure than the old default format of MD5. However, the **md5** option has been added, if you want to see the fingerprint in MD5 format for backward compatibility.

Command or Action	Purpose
switch# **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.
switch# **show ssh server**	Displays the SSH server configuration.
switch# **show user-account**	Displays user account information.
switch# **show users**	Displays the users logged into the device.
switch# **show crypto ca certificates**	Displays the configured certificate chain and associated trustpoint for X.509v3 certificate-based SSH authentication.
switch# **show crypto ca crl** *trustpoint*	Displays the contents of the CRL list of the specified trustpoint for X.509v3 certificate-based SSH authentication.

Default Settings for SSH

The following table lists the default settings for SSH parameters.

Table 1: Default SSH Parameters

Parameters	Default
SSH server	Enabled
SSH server key	RSA key generated with 1024 bits
RSA key bits for generation	1024
Telnet server	Disabled