

SSH Algorithms for Common Criteria Certification

The SSH Algorithms for Common Criteria Certification feature provides the list and order of the algorithms that are allowed for Common Criteria Certification. This module describes how to configure the encryption, Message Authentication Code (MAC), and host key algorithms for a secure shell (SSH) server and client so that SSH connections can be limited on the basis of the allowed algorithms list.

- Restriction for SSH Algorithms for Common Criteria Certification, on page 1
- Information About SSH Algorithms for Common Criteria Certification, on page 2
- How to Configure SSH Algorithms for Common Criteria Certification, on page 4
- Configuration Examples for SSH Algorithms for Common Criteria Certification, on page 9
- Additional References for SSH Algorithms for Common Criteria Certification, on page 10
- Feature Information for SSH Algorithms for Common Criteria Certification, on page 11

Restriction for SSH Algorithms for Common Criteria Certification

- Starting from Cisco IOS XE Release 17.10, the following Key Exchange and MAC algorithms are removed from the default list:

  Key Exchange algorithm:

  - diffie-hellman-group14-sha1

  MAC algorithms:

  - hmac-sha1
  - hmac-sha2-256
  - hmac-sha2-512

Note You can use the **ip ssh server algorithm kex** command to configure the Key Exchange algorithm and the **ip ssh server algorithm mac** command to configure the MAC algorithms.

Information About SSH Algorithms for Common Criteria Certification

SSH Algorithms for Common Criteria Certification

A Secure Shell (SSH) configuration enables a Cisco IOS SSH server and client to authorize the negotiation of only those algorithms that are configured from the allowed list. If a remote party tries to negotiate using only those algorithms that are not part of the allowed list, the request is rejected and the session is not established.

Cisco IOS SSH Server Algorithms

Cisco IOS secure shell (SSH) servers support the encryption algorithms (Advanced Encryption Standard Counter Mode [AES-CTR], AES Cipher Block Chaining [AES-CBC], Triple Data Encryption Standard [3DES]), and Galois/Counter Mode (GCM)), the Message Authentication Code (MAC) algorithms, the host key algorithms, the Key Exchange (KEX) DH Group algorithms, and the public key algorithms in the following order:

Table 1: Supported Default and Non-Default IOS SSH Server Algorithms

Supported Algorithms	Default	Non-Default
Encryption	**1.** chacha20-poly1305@openssh.com **2.** aes128-gcm@openssh.com **3.** aes256-gcm@openssh.com **4.** aes128-gcm **5.** aes256-gcm **6.** aes128-ctr **7.** aes192-ctr **8.** aes256-ctr	• aes128-cbc • aes192-cbc • aes256-cbc • 3des-cbc
HMAC	**1.** hmac-sha2-256-etm@openssh.com **2.** hmac-sha2-512-etm@openssh.com	• hmac-sha1 • hmac-sha2-256 • hmac-sha2-512
Host Key	**1.** rsa-sha2-512 **2.** rsa-sha2-256 **3.** ssh-rsa	• x509v3-ssh-rsa

Supported Algorithms	Default	Non-Default
KEX DH Group	**1.** curve25519-sha256 **2.** curve25519-sha256@libssh.org **3.** ecdh-sha2-nistp256 **4.** ecdh-sha2-nistp384 **5.** ecdh-sha2-nistp521 **6.** diffie-hellman-group14-sha256 **7.** diffie-hellman-group16-sha512	• diffie-hellman-group14-sha1
Public Key	**1.** ssh-rsa **2.** ecdsa-sha2-nistp256 **3.** ecdsa-sha2-nistp384 **4.** ecdsa-sha2-nistp521 **5.** ssh-ed25519 **6.** x509v3-ecdsa-sha2-nistp256 **7.** x509v3-ecdsa-sha2-nistp384 **8.** x509v3-ecdsa-sha2-nistp521 **9.** rsa-sha2-256 **10.** rsa-sha2-512 **11.** x509v3-rsa2048-sha256	• x509v3-ssh-rsa

Cisco IOS SSH Client Algorithms

Cisco IOS secure shell (SSH) clients support the encryption algorithms (Advanced Encryption Standard counter mode [AES-CTR], AES Cipher Block Chaining [AES-CBC], Triple Data Encryption Standard [3DES]), and Galois/Counter Mode (GCM)), the MAC algorithms, and the KEX DH Group algorithms in the following order:

Table 2: Supported Default and Non-Default IOS SSH Server Algorithms

Supported Algorithms	Default	Non-Default
Encryption	1. chacha20-poly1305@openssh.com 2. aes128-gcm@openssh.com 3. aes256-gcm@openssh.com 4. aes128-gcm 5. aes256-gcm 6. aes128-ctr 7. aes192-ctr 8. aes256-ctr	• aes128-cbc • aes192-cbc • aes256-cbc • 3des-cbc
HMAC	1. hmac-sha2-256-etm@openssh.com 2. hmac-sha2-512-etm@openssh.com	• hmac-sha1 • hmac-sha2-256 • hmac-sha2-512
KEX DH Group	1. curve25519-sha256 2. curve25519-sha256@libssh.org 3. ecdh-sha2-nistp256 4. ecdh-sha2-nistp384 5. ecdh-sha2-nistp521 6. diffie-hellman-group14-sha256 7. diffie-hellman-group16-sha512	• diffie-hellman-group14-sha1

How to Configure SSH Algorithms for Common Criteria Certification

Configuring an Encryption Key Algorithm for a Cisco IOS SSH Server and Client

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ssh** {**server** | **client**} **algorithm encryption** {**aes128-ctr** | **aes192-ctr** | **aes256-ctr** | **aes128-cbc** | **3des-cbc** | **aes192-cbc** | **aes256-cbc**}

4. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**ip ssh** {**server** | **client**} **algorithm encryption** {**aes128-ctr** | **aes192-ctr** | **aes256-ctr** | **aes128-cbc** | **3des-cbc** | **aes192-cbc** | **aes256-cbc**}
	Example:
	`Device(config)# ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr aes128-cbc 3des-cbc aes192-cbc aes256-cbc`
	`Device(config)# ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr aes128-cbc 3des-cbc aes192-cbc aes256-cbc`
		Defines the order of encryption algorithms in the SSH server and client. This order is presented during algorithm negotiation.
		Note The Cisco IOS SSH server and client must have at least one configured encryption algorithm.
		Note To disable one algorithm from the previously configured algorithm list, use the **no** form of this command. To disable more than one algorithm, use the **no** form of this command multiple times with different algorithm names.
		Note For a default configuration, use the default form of this command as shown below:
		`Device(config)# ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr aes128-cbc 3des-cbc aes192-cbc aes256-cbc`

Step 4	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

If you try to disable the last encryption algorithm in the configuration, the following message is displayed and the command is rejected:

```
% SSH command rejected: All encryption algorithms cannot be disabled
```

Configuring a MAC Algorithm for a Cisco IOS SSH Server and Client

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**ip ssh** {**server** | **client**} **algorithm mac** {**hmac-sha2** | **hmac-sha2-96**}
	Example:
	`Device(config)# ip ssh server algorithm mac hmac-sha2 hmac-sha2-96`
	`Device(config)# ip ssh client algorithm mac hmac-sha2 hmac-sha2-96`
		Defines the order of MAC (Message Authentication Code) algorithms in the SSH server and client. This order is presented during algorithm negotiation.
		Note
		The Cisco IOS SSH server and client must have at least one configured Hashed Message Authentication Code (HMAC) algorithm.
		Note
		To disable one algorithm from the previously configured algorithm list, use the **no** form of this command. To disable more than one algorithm, use the **no** form of this command multiple times with different algorithm names.
		Note
		For default configuration, use the default form of this command as shown below:
		`Device(config)# ip ssh server algorithm mac hmac-sha2 hmac-sha2-96`

Step 4	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

If you try to disable the last MAC algorithm in the configuration, the following message is displayed and the command is rejected:

```
% SSH command rejected: All mac algorithms cannot be disabled
```

Configuring a Host Key Algorithm for a Cisco IOS SSH Server

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ssh server algorithm hostkey** {**x509v3-ssh-rsa** | **ssh-rsa**}
4. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip ssh server algorithm hostkey** {**x509v3-ssh-rsa**	**ssh-rsa**} **Example:** `Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa ssh-rsa`	Defines the order of host key algorithms. Only the configured algorithm is negotiated with the Cisco IOS secure shell (SSH) client. **Note** The Cisco IOS SSH server must have at least one configured host key algorithm: • x509v3-ssh-rsa—X.509v3 certificate-based authentication • ssh-rsa—Public-key-based authentication **Note** To disable one algorithm from the previously configured algorithm list, use the **no** form of this command. To disable

	Command or Action	Purpose
		more than one algorithm, use the **no** form of this command multiple times with different algorithm names. **Note** For default configuration, use the default form of this command as shown below: `Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa ssh-rsa`
Step 4	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

If you try to disable the last host key algorithm in the configuration, the following message is displayed and the command is rejected:

```
% SSH command rejected: All hostkey algorithms cannot be disabled
```

Verifying SSH Algorithms for Common Criteria Certification

SUMMARY STEPS

1. **enable**
2. **show ip ssh**

DETAILED STEPS

Procedure

Step 1 **enable**

Enables privileged EXEC mode.

- Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 **show ip ssh**

Displays configured Secure Shell (SSH) encryption, host key, and Message Authentication Code (MAC) algorithms.

Example:

The following sample output from the **show ip ssh** command shows the encryption algorithms configured in the default order:

```
Device# show ip ssh

Encryption Algorithms: aes128-ctr, aes192-ctr, aes256-ctr, aes128-cbc, 3des-cbc, aes192-cbc, aes256-cbc
```

The following sample output from the **show ip ssh** command shows the MAC algorithms configured in the default order:

```
Device# show ip ssh

MAC Algorithms: hmac-sha1 hmac-sha1-96
```

The following sample output from the **show ip ssh** command shows the host key algorithms configured in the default order:

```
Device# show ip ssh

Hostkey Algorithms: x509v3-ssh-rsa, ssh-rsa
```

Configuration Examples for SSH Algorithms for Common Criteria Certification

Example: Configuring Encryption Key Algorithms for a Cisco IOS SSH Server

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr aes128-cbc
 3des-cbc aes192-cbc aes256-cbc
Device(config)# end
```

Example: Configuring Encryption Key Algorithms for a Cisco IOS SSH Client

```
Device> enable
Device# configure terminal
Device(config)# ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr aes128-cbc
 3des-cbc aes192-cbc aes256-cbc
Device(config)# end
```

Example: Configuring MAC Algorithms for a Cisco IOS SSH Server

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm mac hmac-sha1 hmac-sha1-96
Device(config)# end
```

Example: Configuring Key Exchange DH Group for a Cisco IOS SSH Server

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm kex diffie-hellman-group-exchange-sha1
Device(config)# end
```

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm kex diffie-hellman-group14-sha1
Device(config)# end
```

Example: Configuring Host Key Algorithms for a Cisco IOS SSH Server

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa ssh-rsa
Device(config)# end
```

Additional References for SSH Algorithms for Common Criteria Certification

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Security commands	• Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

Related Topic	Document Title
SSH authentication	"Secure Shell-Configuring User Authentication Methods" chapter in the *Secure Shell Configuration Guide*
X.509v3 digital certificates in server and user authentication	"X.509v3 Certificates for SSH Authentication" chapter in the *Secure Shell Configuration Guide*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for SSH Algorithms for Common Criteria Certification

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 3: Feature Information for SSH Algorithms for Common Criteria Certification

Feature Name	Releases	Feature Information
SSH Algorithms for Common Criteria Certification	Cisco IOS XE Everest 16.5.1a	The SSH Algorithms for Common Criteria Certification feature provides the list and order of the algorithms that are allowed for Common Criteria Certification. This module describes how to configure the encryption, Message Authentication Code (MAC), and host key algorithms for a secure shell (SSH) server and client so that SSH connections can be limited on the basis of the allowed algorithms list. The following commands were introduced by this feature: **ip ssh** {**server** | **client**} **algorithm encryption**, **ip ssh** {**server** | **client**} **algorithm mac**.
SSH Algorithms for Common Criteria Certification	Cisco IOS XE Cupertino 17.8.1	Cisco IOS SSH Server and Client support for the following algorithms have been introduced: • chacha20-poly1305@openssh.com • ssh-ed25519 • curve25519-sha256@libssh.org
SSH Algorithms for Common Criteria Certification	Cisco IOS XE Cupertino 17.9.1	Cisco IOS SSH Server and Client support for the following algorithms have been introduced: • aes128-gcm@openssh.com • aes256-gcm@openssh.com

Deprecation of Weak Ciphers	Cisco IOS XE Release 17.10	The following changes have been introduced:
		• The Secure Shell Version 1.99 is not supported.
		• The following weak Key Exchange and MAC algorithms are removed from the default list of algorithms:
		  • diffie-hellman-group14-sha1
		  • hmac-sha1
		  • hmac-sha2-256
		  • hmac-sha2-512

Feature Name	Releases	Feature Information
SSH Algorithms for Common Criteria Certification	Cisco IOS XE Release 17.11.1a	Cisco IOS SSH Server and Client support for the following algorithms have been introduced: • curve25519-sha256 • diffie-hellman-group14-sha256 • diffie-hellman-group16-sha512 • x509v3-rsa2048-sha256