Configuring Certificate Authentication for SSH on Cisco IOS XE Devices

Contents

Introduction

This document outlines the configuration of Secure Shell (SSH) on Cisco IOS® XE devices using X.509v3 certificates for authentication, in accordance with the guidelines defined in RFC 6187.

Prerequisites

Requirements

Cisco recommends that you have knowledge of the Public Key Infrastructure (PKI) infrastructure.

Components Used

The information in this document is based on these software and hardware versions:

- C9200L Switch running Cisco IOS XE version 17.3.5
- Pragma Fortress SSH client

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command

Background Information

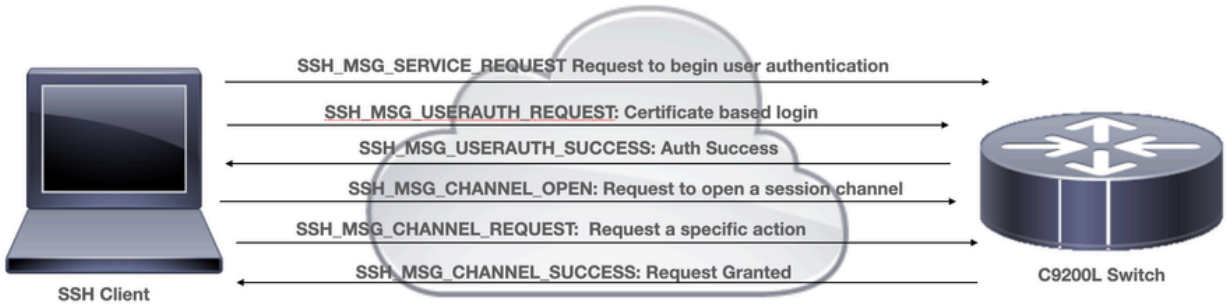
The approach enhances SSH security by enabling certificate-based authentication, thereby aligning SSH key management practices more closely with those used in Transport Layer Security (TLS).

SSH provides mutual authentication in order to establish a secure connection between client and server. Traditionally, servers authenticate using Rivest-Shamir-Addleman (RSA) key pairs. The client computes a fingerprint of the public key of the server and prompts the administrator to verify it — ideally by comparing it with a known value obtained via a secure, out-of-band method. However, this manual verification is often skipped due to its complexity, increasing the risk of man-in-the-middle (MitM) attacks and weakening the SSH trust model.

RFC 6187 addresses these issues by enabling X.509v3 certificate-based authentication, which integrates SSH with PKI. This approach improves security and scalability by allowing trust to be established through trusted Certificate Authorities (CAs), offering a user experience and trust model similar to TLS.

Configure

Network Diagram



Deployment Considerations

- An RFC6187-compatible SSH client is necessary to take advantage of the feature.
- The SSH client and server negotiates supported authentication mechanisms. All of the authentication mechanisms previously supported on the device can continue to run concurrently with x509-based authentication mechanisms in order to ensure smooth transition.
- Administrator can choose to use x509-based authentication method for server only, client only or both.
- In order to successfully verify the authentication data of the other party, the client and server only need to trust a common CA. This means that only the certificate of CA that signed the router certificate must be installed on the client device trusted certificate store.
- The certificate provides information about identity of the other party (Common Name and Subject Alternative Name are typically used for that purpose). The client must compare the hostname or IP address name of the server that was provided as input by the administrator with the identity data available in the presented certificate. It severely limits the opportunities of MitM or other impersonation attacks.

Configurations

IOS-XE Device (SSH Server)

Configure a trustpoint that holds the CA certificate and optionally the router certificate.

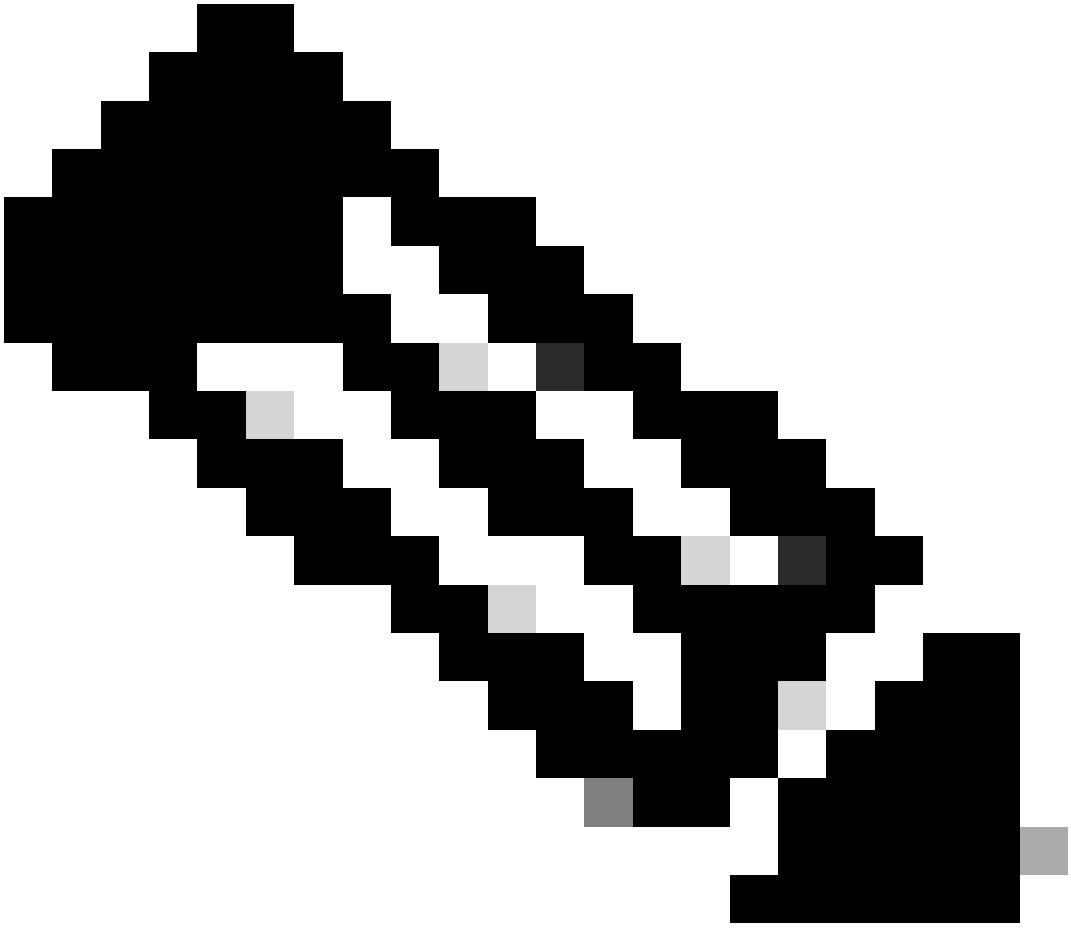
```
crypto pki trustpoint pki-server

 enrollment pkcs12

 subject-name cn=RTR-DC01.cisco.com

 revocation-check none

 rsakeypair ssh-cert

! The username has to be fetched from the certificate for accounting and authorization purposes. Multip

 authorization username subjectname commonname
```

Configure-allowed authentication mechanisms used during SSH tunnel negotiation.

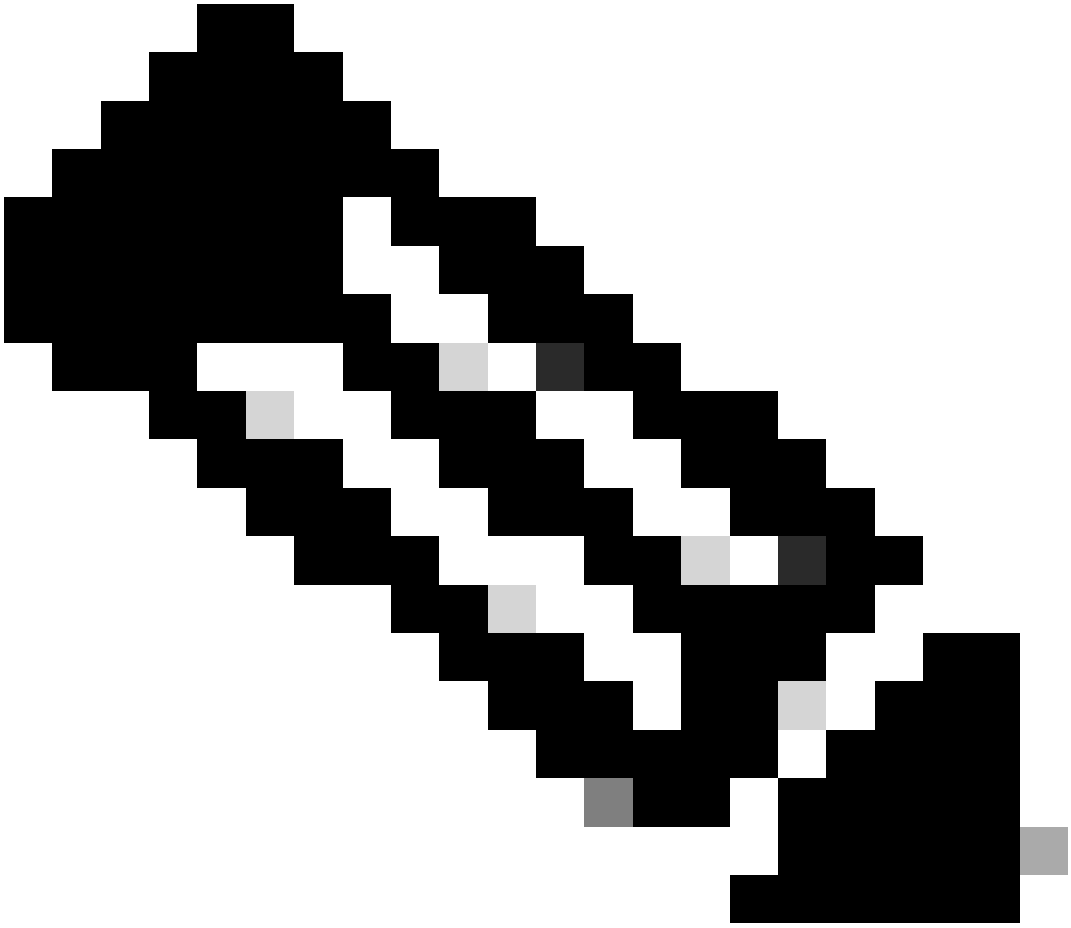
```
! Alorithms used to authenticate server
ip ssh server algorithm hostkey x509v3-ssh-rsa ssh-rsa

! Acceptable algorithms used to authenticate the client
ip ssh server algorithm authentication publickey password keyboard

! Acceptable pubkey-based algorithms used to authenticate the client
ip ssh server algorithm publickey x509v3-ssh-rsa ssh-rsa
```

Configure the SSH server in order to use correct certificates in the authentication process.

```
ip ssh server certificate profile
server
  trustpoint sign <Server-Trustpoint-name>
user
  trustpoint verify <Client-Trustpoint-name>
```

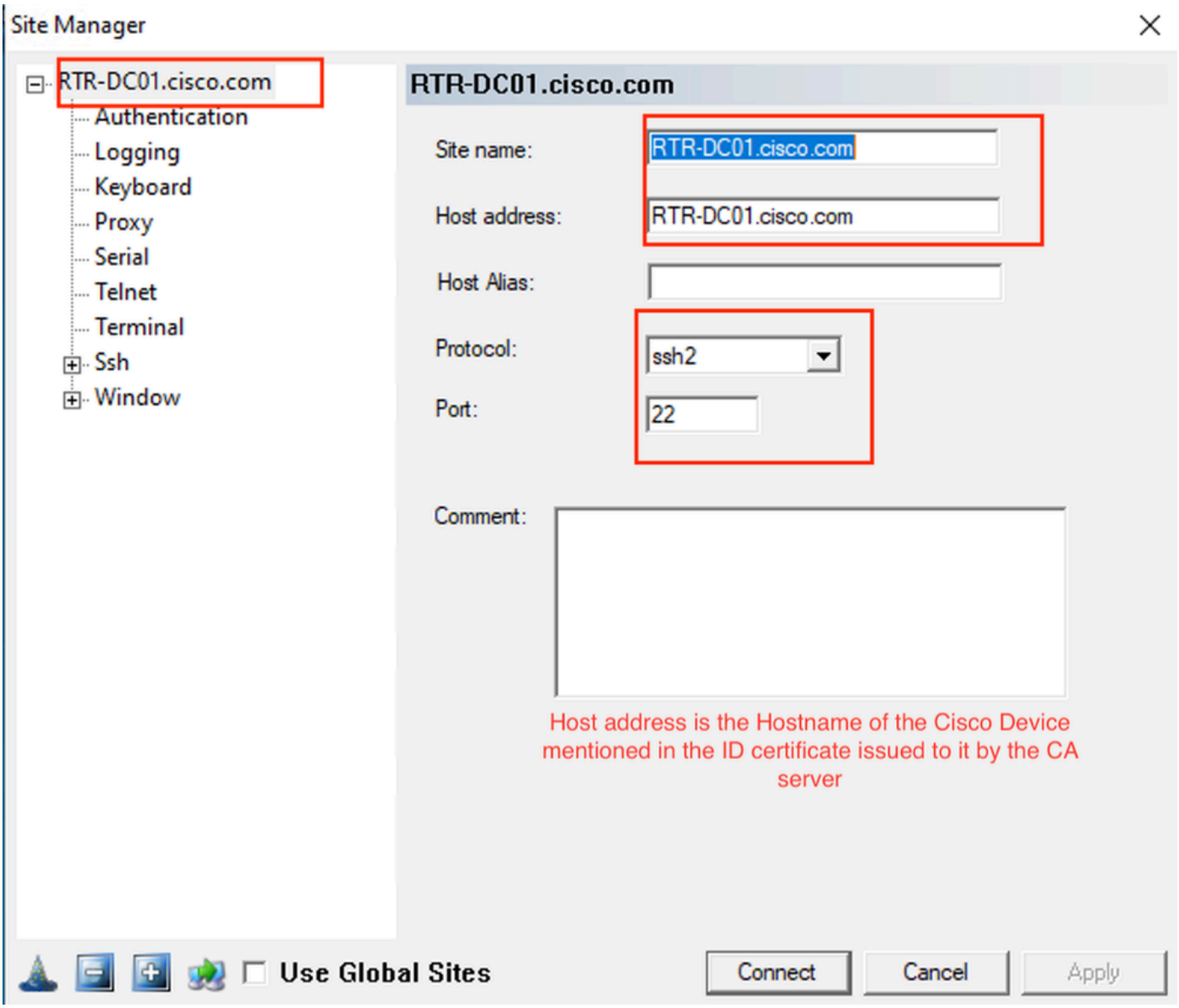


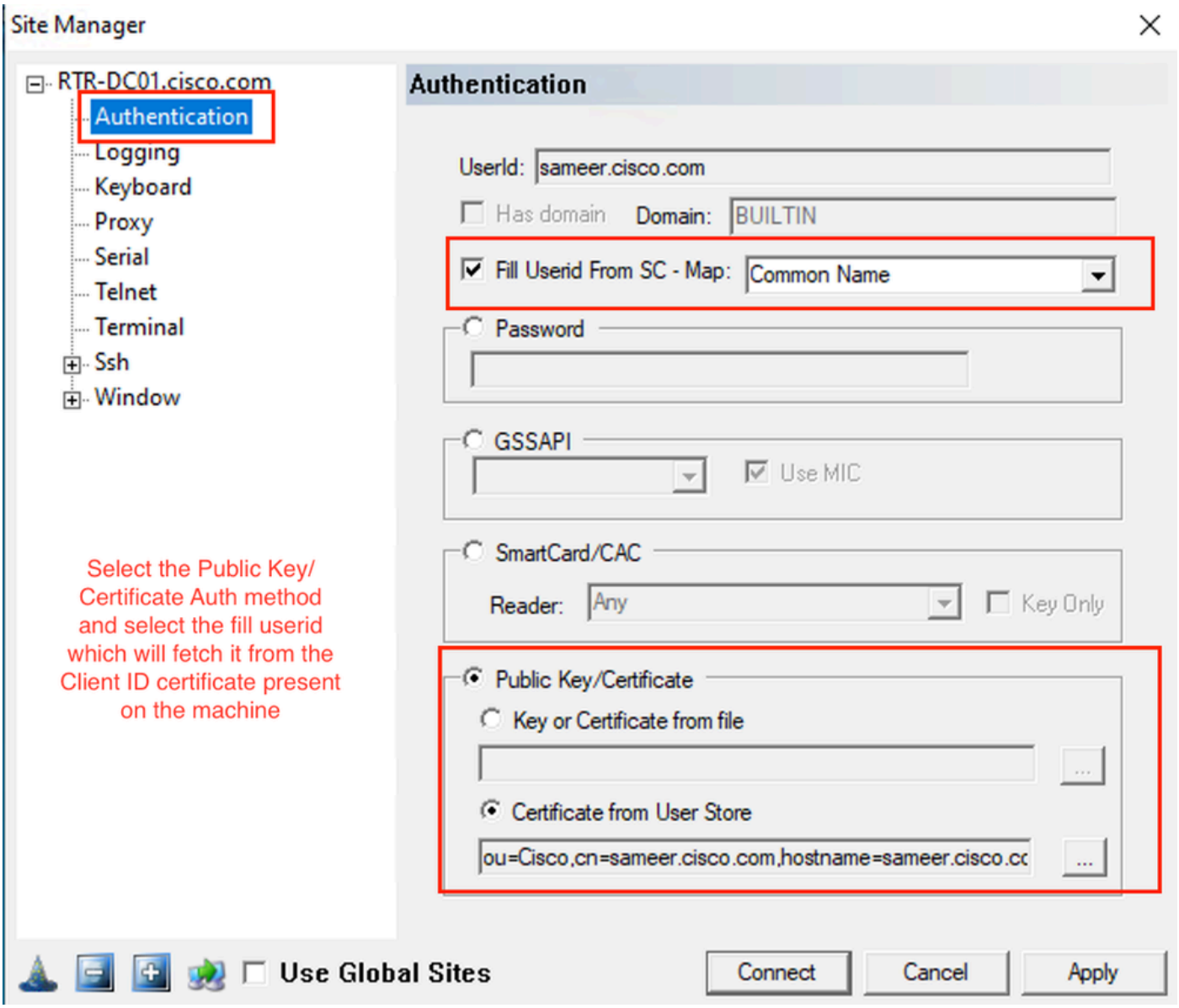
Note: Ensure that the SSH Server and the SSH client have the ID certificate issued by the same CA server.



Note: If in case SSH client such as Windows machine has a ID Certificate issued by some other CA, then have it imported on the SSH server, that is, Cisco Switch and map that Trustpoint to above SSH server certificate profile under user section.

Pragma Fortress CL (SSH Client Installed on User Machine)


Site Manager
RTR-DC01.cisco.com
Authentication
Logging
Keyboard
Proxy
Serial
Telnet
Terminal
Ssh
Window
RTR-DC01.cisco.com
Site name:
RTR-DC01.cisco.com
Host address:
RTR-DC01.cisco.com
Host Alias:
Protocol:
ssh2
Port:
22
Comment:
Host address is the Hostname of the Cisco Device mentioned in the ID certificate issued to it by the CA server
Use Global Sites
Connect
Cancel
Apply



Verify

```
show ip ssh
SSH Enabled - version 1.99
Authentication methods:publickey,password,keyboard-interactive
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa
Hostkey Algorithms:x509v3-ssh-rsa,ssh-rsa
--- output truncated ----

show users
 Line User Host(s) Idle Location
 1 vty 0 sameer.cisco.com idle 00:02:37 192.168.1.100
```

Troubleshoot

These debugs are used in order to track successful session:

```
debug ip ssh detail
debug crypto pki transactions
debug crypto pki messages
debug crypto pki validation

Mar 27 15:35:40.103: SSH1: starting SSH control process
! Server identifies itself
Mar 27 15:35:40.103: SSH1: sent protocol version id SSH-1.99-Cisco-1.25
! Client identifies itself
Mar 27 15:35:40.106: SSH1: protocol version id is - SSH-2.0-Pragma FortressCL 5.0.10.4176

! Authentication algorithms supported by server
Mar 27 15:35:40.106: SSH2 1: kexinit sent: kex algo = diffie-hellman-group-exchange-sha1,diffie-hellman
Mar 27 15:35:40.106: SSH2 1: kexinit sent: hostkey algo = x509v3-ssh-rsa,ssh-rsa
Mar 27 15:35:40.106: SSH2 1: kexinit sent: encryption algo = aes128-ctr,aes192-ctr,aes256-ctr
Mar 27 15:35:40.106: SSH2 1: kexinit sent: mac algo = hmac-sha2-256,hmac-sha2-512,hmac-sha1,hmac-sha1-9
Mar 27 15:35:40.106: SSH2 1: SSH2_MSG_KEXINIT sent
Mar 27 15:35:40.109: SSH2 1: SSH2_MSG_KEXINIT received
Mar 27 15:35:40.109: SSH2 1: kex: client->server enc:aes256-ctr mac:hmac-sha2-256
Mar 27 15:35:40.109: SSH2 1: kex: server->client enc:aes256-ctr mac:hmac-sha2-256

! Client chooses authentication algorithm
Mar 27 15:35:40.109: SSH2 1: Using hostkey algo = x509v3-ssh-rsa
Mar 27 15:35:40.109: SSH2 1: Using kex_algo = diffie-hellman-group-exchange-sha1
Mar 27 15:35:40.109: SSH2 1: SSH2_MSG_KEX_DH_GEX_REQUEST received
Mar 27 15:35:40.109: SSH2 1: Range sent by client is - 1024 < 2048 < 8192
Mar 27 15:35:40.109: SSH2 1:  Modulus size established : 2048 bits
Mar 27 15:35:40.121: SSH2 1: expecting SSH2_MSG_KEX_DH_GEX_INIT
Mar 27 15:35:40.121: SSH2 1: SSH2_MSG_KEXDH_INIT received

! SSH Server sends certificate associated with trustpoint "pki-server"
Mar 27 15:35:40.133: SSH2 1: Sending Server certificate associated with PKI trustpoint "pki-server"
Mar 27 15:35:40.133: SSH2 1: Got 2 certificate(s) on certificate chain
Mar 27 15:35:40.135: SSH2: kex_derive_keys complete
Mar 27 15:35:40.135: SSH2 1: SSH2_MSG_NEWKEYS sent
Mar 27 15:35:40.135: SSH2 1: waiting for SSH2_MSG_NEWKEYS
Mar 27 15:35:40.214: SSH2 0: channel window adjust message received 49926
Mar 27 15:35:41.417: SSH2 1: SSH2_MSG_NEWKEYS received
Mar 27 15:35:41.436: SSH2 1: Authentications that can continue = publickey,password,keyboard-interactiv
Mar 27 15:35:41.437: SSH2 1: Using method = none
Mar 27 15:35:41.437: SSH2 1: Authentications that can continue = publickey,password,keyboard-interactiv
Mar 27 15:35:41.438: SSH2 1: Using method = publickey

! Client sends certificate
Mar 27 15:35:41.438: SSH2 1: Received publickey algo = x509v3-ssh-rsa
Mar 27 15:35:41.438: SSH2 1: Verifying certificate for user 'sameer.cisco.com' in SSH2_MSG_USERAUTH_REQ
Mar 27 15:35:41.439: SSH2 1: Verifying certificate for user 'sameer.cisco.com'
Mar 27 15:35:41.439: SSH2 1: Received a chain of 2 certificate
Mar 27 15:35:41.439: SSH2 1: Received 0 ocsp-response
Mar 27 15:35:41.439: SSH2 1: Starting PKI session for certificate verification

! Client certificate is verified by the SSH-Server
Mar 27 15:35:41.444: SSH2 1: Verifying certificate for user 'sameer.cisco.com'
Mar 27 15:35:41.444: SSH2 1: Received a chain of 2 certificate
Mar 27 15:35:41.444: SSH2 1: Received 0 ocsp-response
Mar 27 15:35:41.444: SSH2 1: Starting PKI session for certificate verification
Mar 27 15:35:41.445: SSH2 1: Verifying signature for user 'sameer.cisco.com' in SSH2_MSG_USERAUTH_REQUE
Mar 27 15:35:41.445: SSH2 1: Received a chain of 2 certificate
Mar 27 15:35:41.445: SSH2 1: Received 0 ocsp-response

! Certificate status verified successfully
Mar 27 15:35:41.446: SSH2 1: Client Signature verification PASSED
```

```
Mar 27 15:35:41.446: SSH2 1: Certificate authentication passed for user 'sameer.cisco.com'
Mar 27 15:35:41.446: SSH2 1: authentication successful for sameer.cisco.com
Mar 27 15:35:41.448: SSH2 1: channel open request
Mar 27 15:35:41.451: SSH2 1: pty-req request
Mar 27 15:35:41.451: SSH2 1: setting TTY - requested: height 25, width 80; set: height 25, width 80
Mar 27 15:35:41.452: SSH2 1: shell request
Mar 27 15:35:41.452: SSH2 1: shell message received
Mar 27 15:35:41.452: SSH2 1: starting shell for vty
Mar 27 15:35:41.464: SSH2 1: channel window adjust message received 9
Aug 21 20:07:32.311: CRYPTO_PKI: ip-ext-val: IP extension validation not required
```

Common Issues

Configuration Mistakes

Certificate validation is successful for the user, however, due to the missing mandatory command **authorization username subjectname commonname** in the configuration, the Certificate Authentication for the user fails.

```
Apr 26 01:35:32.222: SSH1: starting SSH control process
Apr 26 01:35:32.222: SSH1: sent protocol version id SSH-1.99-Cisco-1.25
Apr 26 01:35:32.224: SSH1: protocol version id is - SSH-2.0-Pragma FortressCL 5.0.10.4176
Apr 26 01:35:32.224: SSH2 1: kexinit sent: kex algo = diffie-hellman-group-exchange-sha1,diffie-hellman
Apr 26 01:35:32.224: SSH2 1: kexinit sent: hostkey algo = x509v3-ssh-rsa,ssh-rsa
Apr 26 01:35:32.224: SSH2 1: kexinit sent: encryption algo = aes128-ctr,aes192-ctr,aes256-ctr
Apr 26 01:35:32.224: SSH2 1: kexinit sent: mac algo = hmac-sha2-256,hmac-sha2-512,hmac-sha1,hmac-sha1-9
Apr 26 01:35:32.224: SSH2 1: SSH2_MSG_KEXINIT sent
Apr 26 01:35:32.234: SSH2 1: SSH2_MSG_KEXINIT received
Apr 26 01:35:32.234: SSH2 1: kex: client->server enc:aes256-ctr mac:hmac-sha2-256
Apr 26 01:35:32.235: SSH2 1: kex: server->client enc:aes256-ctr mac:hmac-sha2-256
Apr 26 01:35:32.235: SSH2 1: Using hostkey algo = x509v3-ssh-rsa
Apr 26 01:35:32.235: SSH2 1: Using kex_algo = diffie-hellman-group-exchange-sha1
Apr 26 01:35:32.235: SSH2 1: SSH2_MSG_KEX_DH_GEX_REQUEST received
Apr 26 01:35:32.235: SSH2 1: Range sent by client is - 1024 < 2048 < 8192
Apr 26 01:35:32.235: SSH2 1:  Modulus size established : 2048 bits
Apr 26 01:35:32.246: SSH2 1: expecting SSH2_MSG_KEX_DH_GEX_INIT
Apr 26 01:35:32.246: SSH2 1: SSH2_MSG_KEXDH_INIT received
Apr 26 01:35:32.259: SSH2 1: Sending Server certificate associated with PKI trustpoint “pki-server”
Apr 26 01:35:32.259: CRYPTO_PKI: (A0049) Session started - identity selected (pki-server)
Apr 26 01:35:32.259: SSH2 1: Got 3 certificate(s) on certificate chain
Apr 26 01:35:32.259: CRYPTO_PKI: Rcvd request to end PKI session A0049.
Apr 26 01:35:32.260: CRYPTO_PKI: PKI session A0049 has ended. Freeing all resources.
Apr 26 01:35:32.260: CRYPTO_PKI: unlocked trustpoint pki-server, refcount is 0
Apr 26 01:35:32.273: SSH2: kex_derive_keys complete
Apr 26 01:35:32.274: SSH2 1: SSH2_MSG_NEWKEYS sent
Apr 26 01:35:32.274: SSH2 1: waiting for SSH2_MSG_NEWKEYS
Apr 26 01:35:45.664: SSH2 1: SSH2_MSG_NEWKEYS received
Apr 26 01:35:45.665: SSH2 1: Authentications that can continue = publickey,password,keyboard-interactiv
Apr 26 01:35:45.666: SSH2 1: Using method = none
Apr 26 01:35:45.666: SSH2 1: Authentications that can continue = publickey,password,keyboard-interactiv
Apr 26 01:35:45.675: SSH2 1: Using method = publickey
Apr 26 01:35:45.675: SSH2 1: Received publickey algo = x509v3-ssh-rsa
Apr 26 01:35:45.676: SSH2 1: Verifying certificate for user 'sameer.cisco.com' in SSH2_MSG_USERAUTH_REQ
Apr 26 01:35:45.676: SSH2 1: Verifying certificate for user 'sameer.cisco.com'
Apr 26 01:35:45.676: SSH2 1: Received a chain of 3 certificate
```

```
Apr 26 01:35:45.676: SSH2 1: Received 0 ocsp-response
Apr 26 01:35:45.676: SSH2 1: Starting PKI session for certificate verification
Apr 26 01:35:45.676: CRYPTO_PKI: (A004A) Session started - identity not specified
Apr 26 01:35:45.676: CRYPTO_PKI: (A004A) Adding peer certificate
Apr 26 01:35:45.676: CRYPTO_PKI: Added x509 peer certificate - (1249) bytes
Apr 26 01:35:45.676: CRYPTO_PKI: (A004A) Adding peer certificate
Apr 26 01:35:45.676: CRYPTO_PKI: Added x509 peer certificate - (1215) bytes
Apr 26 01:35:45.676: CRYPTO_PKI: (A004A) Adding peer certificate
Apr 26 01:35:45.676: CRYPTO_PKI: Added x509 peer certificate - (921) bytes
Apr 26 01:35:45.676: CRYPTO_PKI: ip-ext-val: IP extension validation not required
Apr 26 01:35:45.677: CRYPTO_PKI: create new ca_req_context type PKI_VERIFY_CHAIN_CONTEXT,ident 37
Apr 26 01:35:45.677: CRYPTO_PKI: (A004A)validation path has 1 certs
Apr 26 01:35:45.677: CRYPTO_PKI: (A004A) Check for identical certs
Apr 26 01:35:45.677: CRYPTO_PKI : (A004A) Validating non-trusted cert
Apr 26 01:35:45.677: CRYPTO_PKI: (A004A) Create a list of suitable trustpoints
Apr 26 01:35:45.677: CRYPTO_PKI: Found a issuer match
Apr 26 01:35:45.677: CRYPTO_PKI: (A004A) Suitable trustpoints are: pki-server,
Apr 26 01:35:45.677: CRYPTO_PKI: (A004A) Attempting to validate certificate using pki-server policy
Apr 26 01:35:45.677: CRYPTO_PKI: EKU validation successful. Cert matches configuration.
Apr 26 01:35:45.677: CRYPTO_PKI: (A004A) Using pki-server to validate certificate
Apr 26 01:35:45.677: CRYPTO_PKI: Added 1 certs to trusted chain.
Apr 26 01:35:45.677: CRYPTO_PKI: Prepare session revocation service providers
Apr 26 01:35:45.677:  CRYPTO_PKI: Deleting cached key having key id 50
Apr 26 01:35:45.677:  CRYPTO_PKI: Attempting to insert the peer's public key into cache
Apr 26 01:35:45.677:  CRYPTO_PKI:Peer's public inserted successfully with key id 51
Apr 26 01:35:45.678:  CRYPTO_PKI: Expiring peer's cached key with key id 51
Apr 26 01:35:45.678: CRYPTO_PKI: (A004A) Certificate is verified
Apr 26 01:35:45.678: CRYPTO_PKI: Remove session revocation service providers
Apr 26 01:35:45.678: CRYPTO_PKI: Remove session revocation service providers
Apr 26 01:35:45.678: CRYPTO_PKI: (A004A) Certificate validated without revocation check
Apr 26 01:35:45.678: CRYPTO_PKI: Populate AAA auth data
Apr 26 01:35:45.678: CRYPTO_PKI: Unable to get configured attribute for primary AAA list authorization.
Apr 26 01:35:45.678: CRYPTO_PKI: (A004A)chain cert was anchored to trustpoint pki-server, and chain val
Apr 26 01:35:45.678: CRYPTO_PKI: destroying ca_req_context type PKI_VERIFY_CHAIN_CONTEXT,ident 37, ref
Apr 26 01:35:45.678: CRYPTO_PKI: ca_req_context released
Apr 26 01:35:45.678: CRYPTO_PKI: (A004A) Validation TP is pki-server
Apr 26 01:35:45.678: CRYPTO_PKI: (A004A) Certificate validation succeeded
Apr 26 01:35:45.678: SSH2 1: Could not find username field configured for certificate in trustpoint Err
Apr 26 01:35:45.678: CRYPTO_PKI: Rcvd request to end PKI session A004A.
Apr 26 01:35:45.678: CRYPTO_PKI: PKI session A004A has ended. Freeing all resources.
Apr 26 01:35:45.679: SSH2 1: Can not decode the certificate Err code = 7
Apr 26 01:35:45.679: SSH2 1: Certificate authentication failed for user 'sameer.cisco.com'
```

Related Information

- PKI Configuration Guide
- Cisco Technical Support & Downloads