Configuración de la Autenticación de Certificados para SSH en Dispositivos Cisco IOS XE

Contenido

Introducción

Este documento describe la configuración de Secure Shell (SSH) en los dispositivos Cisco IOS® XE mediante certificados X.509v3 para la autenticación, de acuerdo con las pautas definidas en RFC 6187.

Prerequisites

Requirements

Cisco recomienda que conozca la infraestructura de Public Key Infrastructure (PKI).

Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Switch C9200L que ejecuta Cisco IOS XE versión 17.3.5

- cliente Pragma Fortress SSH

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando

Antecedentes

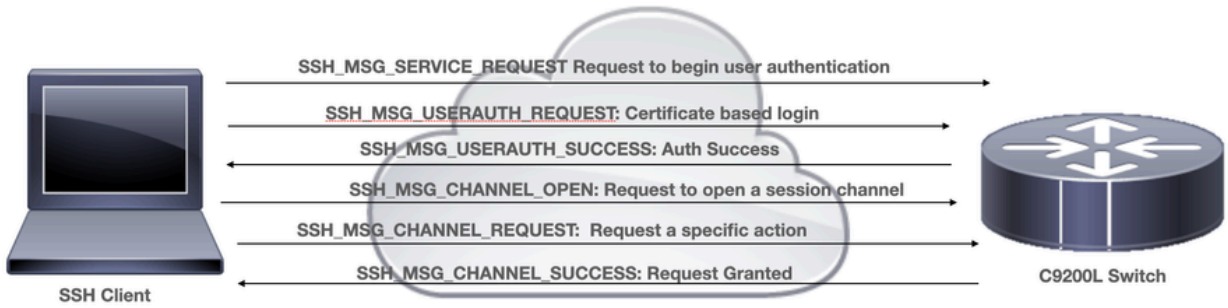
El enfoque mejora la seguridad de SSH al habilitar la autenticación basada en certificados, alineando así las prácticas de administración de claves SSH más estrechamente con las utilizadas en la seguridad de la capa de transporte (TLS).

SSH proporciona autenticación mutua para establecer una conexión segura entre el cliente y el servidor. Tradicionalmente, los servidores se autentican mediante pares de claves RSA (Rivest-Shamir-Addleman). El cliente calcula una huella dactilar de la clave pública del servidor y pide al administrador que la verifique; lo ideal sería compararla con un valor conocido obtenido mediante un método seguro y fuera de banda. Sin embargo, esta verificación manual a menudo se omite debido a su complejidad, lo que aumenta el riesgo de ataques de intrusos (MitM) y debilita el modelo de confianza SSH.

RFC 6187 soluciona estos problemas al habilitar la autenticación basada en certificados X.509v3, que integra SSH con PKI. Este enfoque mejora la seguridad y la escalabilidad al permitir que se establezca la confianza a través de las autoridades de certificación (CA) de confianza, lo que ofrece una experiencia de usuario y un modelo de confianza similares a TLS.

Configurar

Diagrama de la red



Consideraciones de implementación

- Se necesita un cliente SSH compatible con RFC6187 para aprovechar la función.
- El cliente y el servidor SSH negocian los mecanismos de autenticación soportados. Todos los mecanismos de autenticación previamente admitidos en el dispositivo pueden continuar ejecutándose simultáneamente con los mecanismos de autenticación basados en x509 para garantizar una transición sin problemas.
- El administrador puede optar por utilizar un método de autenticación basado en x509 sólo para servidor, sólo para cliente o ambos.
- Para comprobar correctamente los datos de autenticación de la otra parte, el cliente y el servidor sólo necesitan confiar en una CA común. Esto significa que sólo el certificado de la CA que firmó el certificado del router debe estar instalado en el almacén de certificados de confianza del dispositivo cliente.
- El certificado proporciona información sobre la identidad de la otra parte (el nombre común y el nombre alternativo del sujeto se suelen utilizar para ese fin). El cliente debe comparar el nombre de host o el nombre de la dirección IP del servidor proporcionado como entrada por el administrador con los datos de identidad disponibles en el certificado presentado. Limita en gran medida las oportunidades de MitM u otros ataques de suplantación.

Configuraciones

Dispositivo IOS-XE (servidor SSH)

Configure un punto de confianza que contenga el certificado de la CA y, opcionalmente, el certificado del router.

```
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
```

Mecanismos de autenticación permitidos por la configuración utilizados durante la negociación del túnel SSH.

```
! 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
```

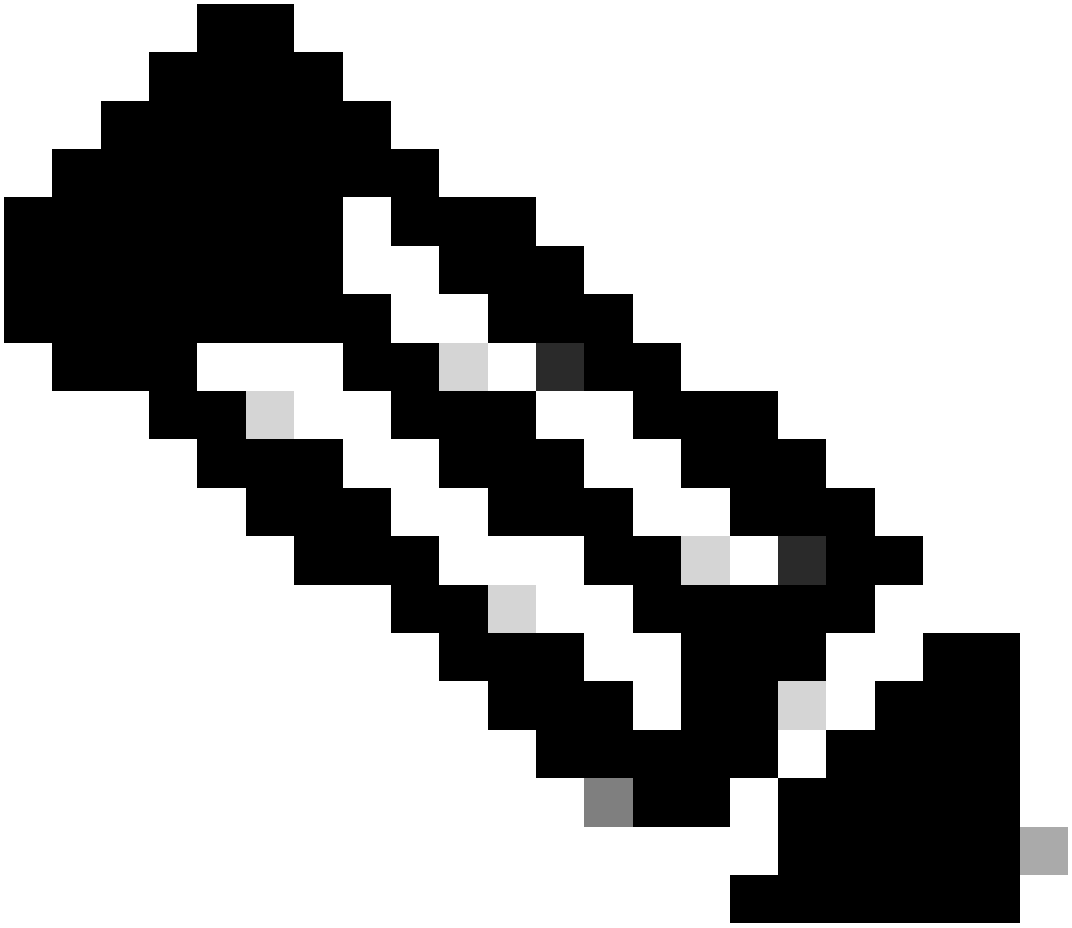
```
! Acceptable pubkey-based algorithms used to authenticate the client
ip ssh server algorithm publickey x509v3-ssh-rsa ssh-rsa
```

Configure el servidor SSH para utilizar los certificados correctos en el proceso de autenticación.

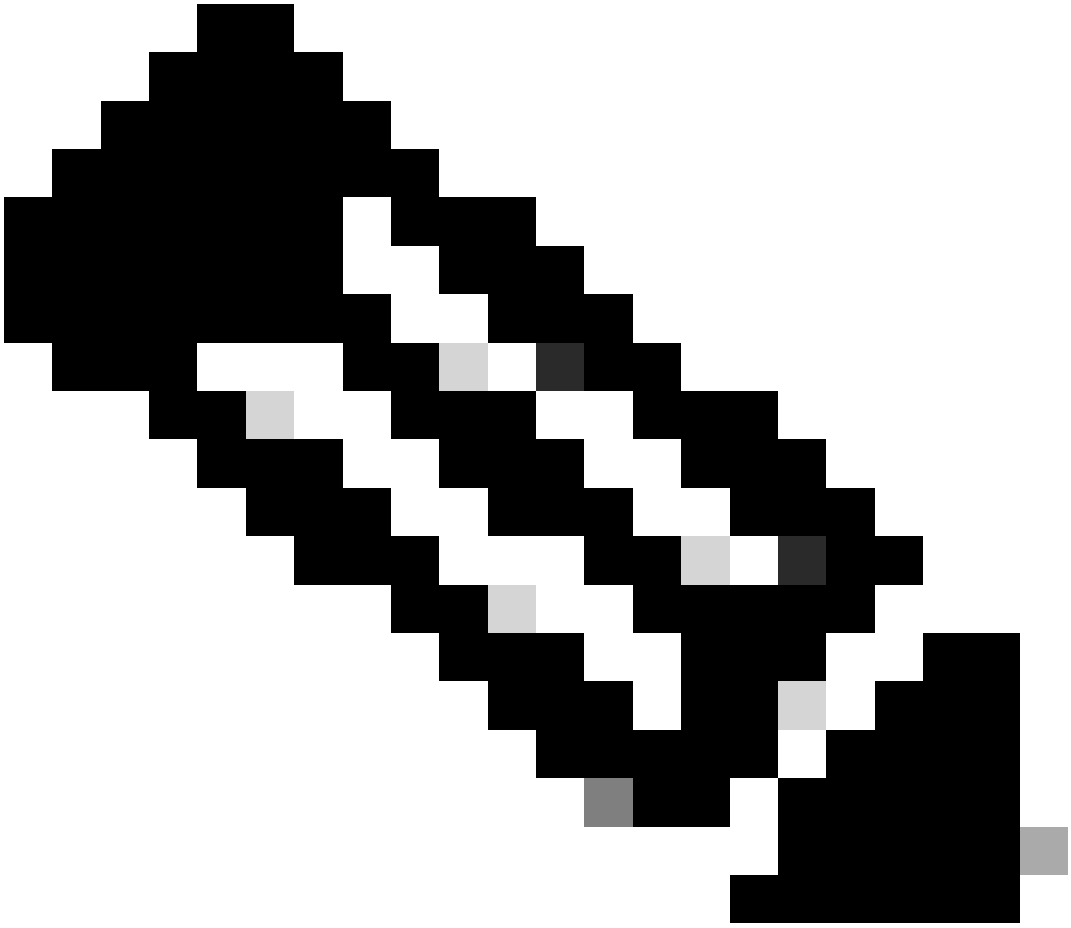
```
ip ssh server certificate profile
server
  trustpoint sign



user
  trustpoint verify
```

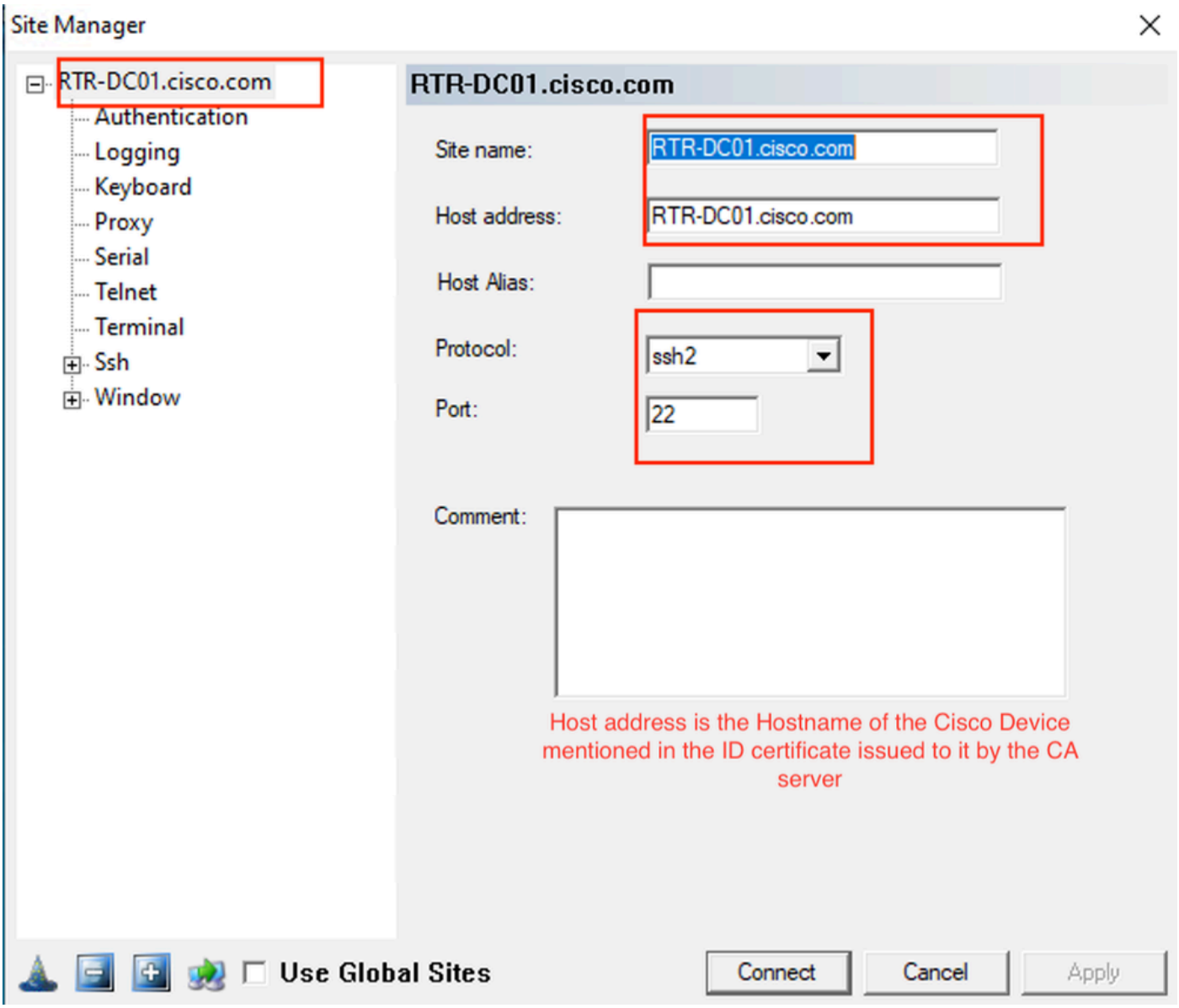


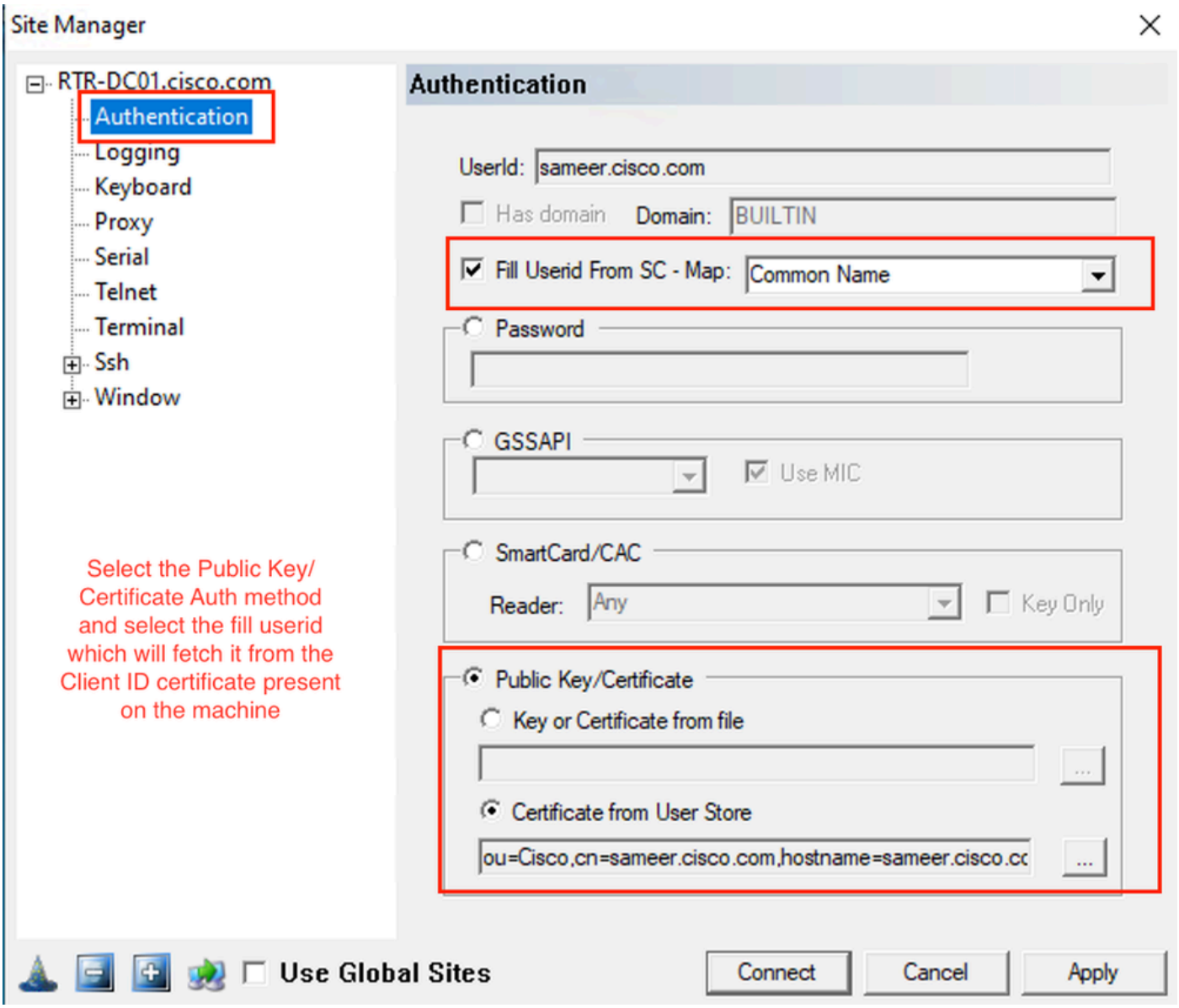
Nota: Asegúrese de que el servidor SSH y el cliente SSH tengan el certificado de ID emitido por el mismo servidor CA.



Nota: Si, en caso de que un cliente SSH como una máquina Windows tenga un Certificado de ID emitido por alguna otra CA, impórtelo en el servidor SSH, es decir, Cisco Switch y asigne ese Trustpoint a un perfil de certificado de servidor SSH superior en la sección de usuario.

Pragma Fortress CL (cliente SSH instalado en la máquina del usuario)


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



Verificación

```
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

Estos debugs se utilizan para realizar un seguimiento de la sesión exitosa:

```
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
```

Problemas comunes

Errores de configuración

La validación del certificado es exitosa para el usuario; sin embargo, debido a que falta el comando obligatorio authorization username subjectname commonname en la configuración, la autenticación del certificado para el usuario falla.

```
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'
```

Información Relacionada

- Guía de configuración de PKI
- Soporte técnico y descargas de Cisco

Acerca de esta traducción
Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).