Configuration de l'authentification de certificat pour SSH sur les périphériques Cisco IOS XE

Table des matières

Introduction

Ce document décrit la configuration de Secure Shell (SSH) sur les périphériques Cisco IOS® XE à l'aide de certificats X.509v3 pour l'authentification, conformément aux directives définies dans la RFC 6187.

Conditions préalables

Exigences

Cisco vous recommande de connaître l'infrastructure PKI (Public Key Infrastructure).

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Commutateur C9200L exécutant Cisco IOS XE version 17.3.5
- Client SSH Pragma Fortress

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. Si votre réseau est en ligne, assurez-vous de bien comprendre l’incidence possible des commandes

Informations générales

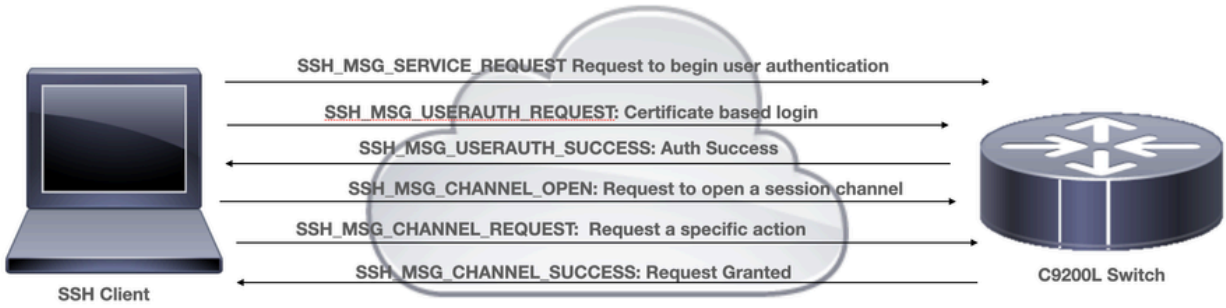
L'approche améliore la sécurité SSH en activant l'authentification basée sur les certificats, alignant ainsi les pratiques de gestion des clés SSH plus étroitement avec celles utilisées dans la sécurité de la couche transport (TLS).

SSH fournit une authentification mutuelle afin d'établir une connexion sécurisée entre le client et le serveur. Traditionnellement, les serveurs s'authentifient à l'aide de paires de clés Rivest-Shamir-Addleman (RSA). Le client calcule une empreinte de la clé publique du serveur et invite l'administrateur à la vérifier, idéalement en la comparant à une valeur connue obtenue via une méthode hors bande sécurisée. Cependant, cette vérification manuelle est souvent ignorée en raison de sa complexité, augmentant le risque d'attaques de l'homme du milieu (MitM) et affaiblissant le modèle de confiance SSH.

RFC 6187 résout ces problèmes en activant l'authentification basée sur certificat X.509v3, qui intègre SSH avec PKI. Cette approche améliore la sécurité et l'évolutivité en permettant d'établir la confiance par le biais d'autorités de certification (CA) fiables, offrant une expérience utilisateur et un modèle de confiance similaires à TLS.

Configurer

Diagramme du réseau



Considérations de déploiement

- Un client SSH compatible RFC6187 est nécessaire pour tirer parti de cette fonctionnalité.
- Le client et le serveur SSH négocient les mécanismes d'authentification pris en charge. Tous les mécanismes d'authentification précédemment pris en charge sur le périphérique peuvent

continuer à fonctionner simultanément avec les mécanismes d'authentification x509 afin d'assurer une transition en douceur.
- L'administrateur peut choisir d'utiliser la méthode d'authentification basée sur x509 pour le serveur uniquement, le client uniquement ou les deux.
- Afin de vérifier avec succès les données d'authentification de l'autre partie, le client et le serveur doivent seulement faire confiance à une autorité de certification commune. Cela signifie que seul le certificat de l'autorité de certification qui a signé le certificat du routeur doit être installé sur le magasin de certificats de confiance du périphérique client.
- Le certificat fournit des informations sur l'identité de l'autre partie (le nom commun et le nom alternatif du sujet sont généralement utilisés à cette fin). Le client doit comparer le nom d'hôte ou le nom d'adresse IP du serveur fourni en entrée par l'administrateur avec les données d'identité disponibles dans le certificat présenté. Elle limite considérablement les possibilités d'attaques par MitM ou par usurpation d'identité.

Configurations

Périphérique IOS-XE (serveur SSH)

Configurez un point de confiance qui contient le certificat CA et éventuellement le certificat du routeur.

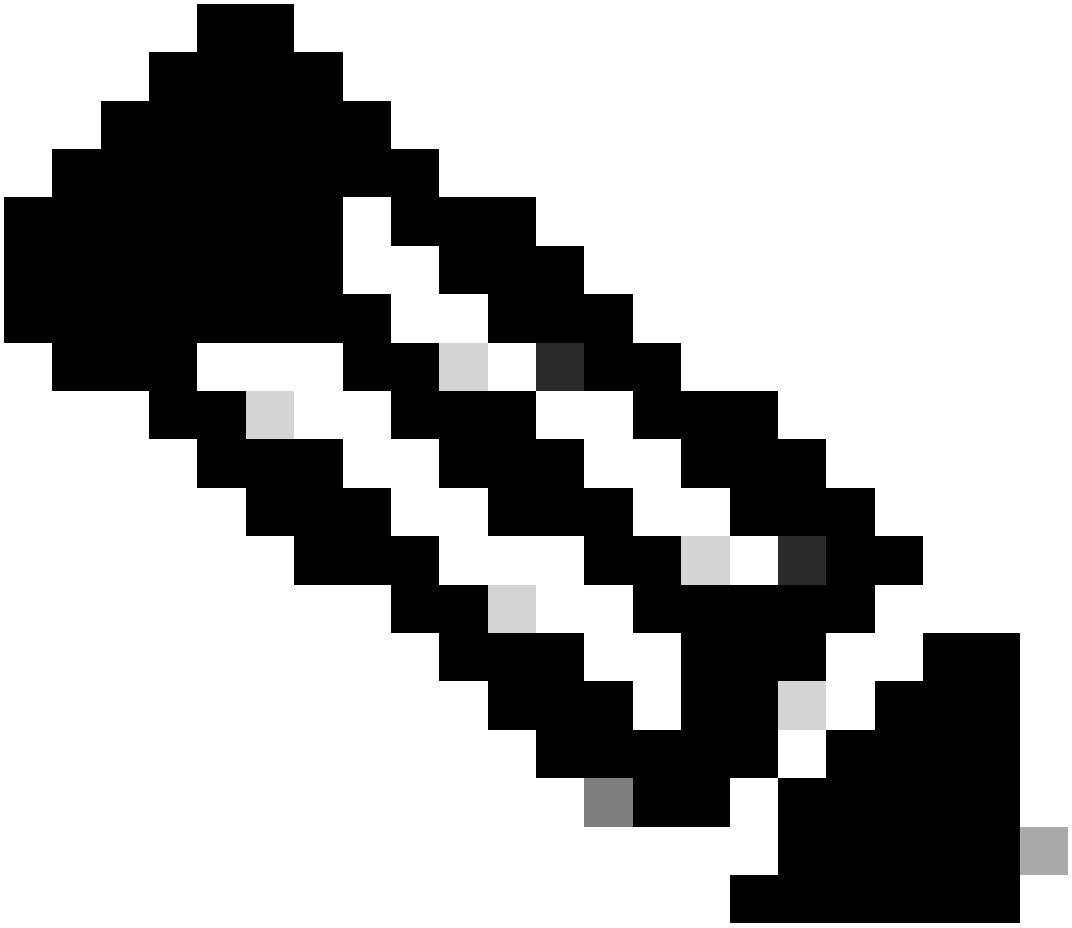
```
crypto pki trustpoint pki-server

 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
```

Configurez les mécanismes d'authentification autorisés utilisés lors de la négociation du tunnel SSH.

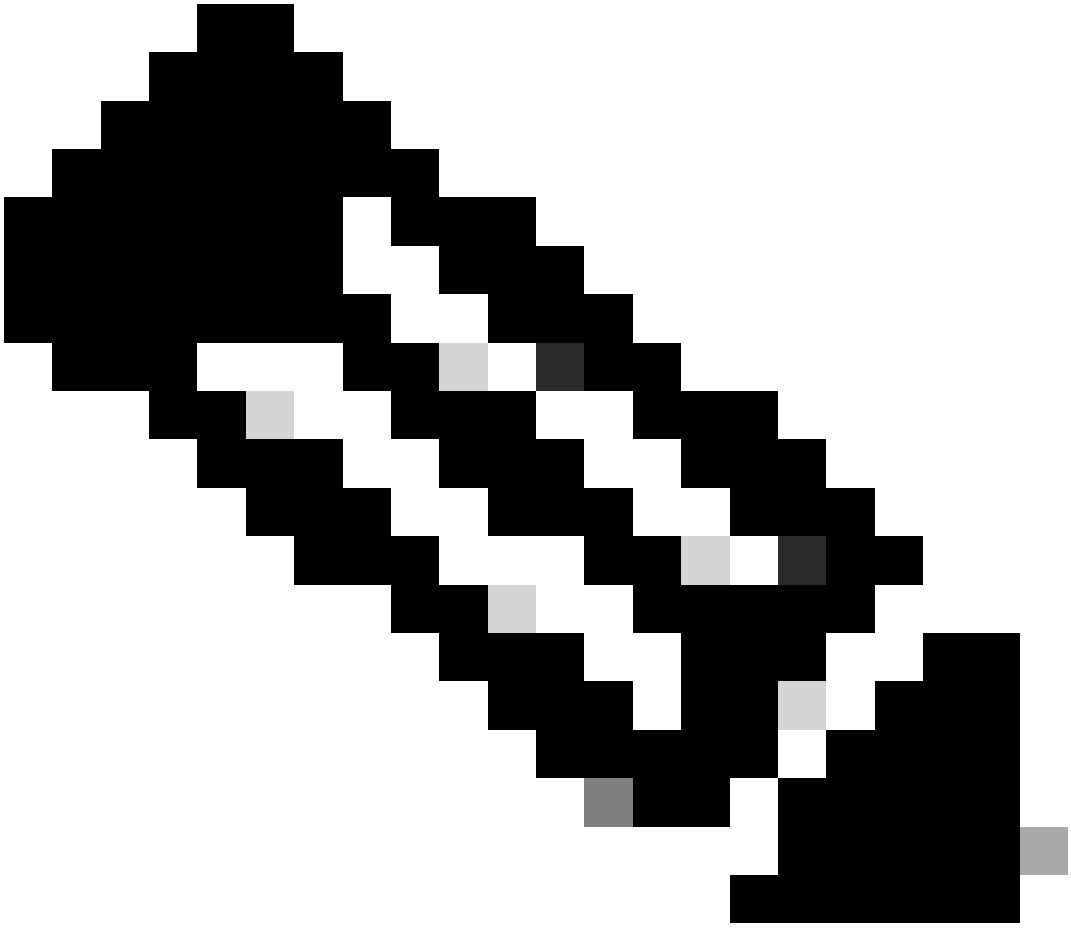
```
! Alorithms used to authenticate server
ip ssh server algorithm hostkey x509v3-ssh-rsa ssh-rsa

! 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
```

Configurez le serveur SSH afin d'utiliser les certificats corrects dans le processus d'authentification.

```
ip ssh server certificate profile
server
  trustpoint sign



user
  trustpoint verify
```

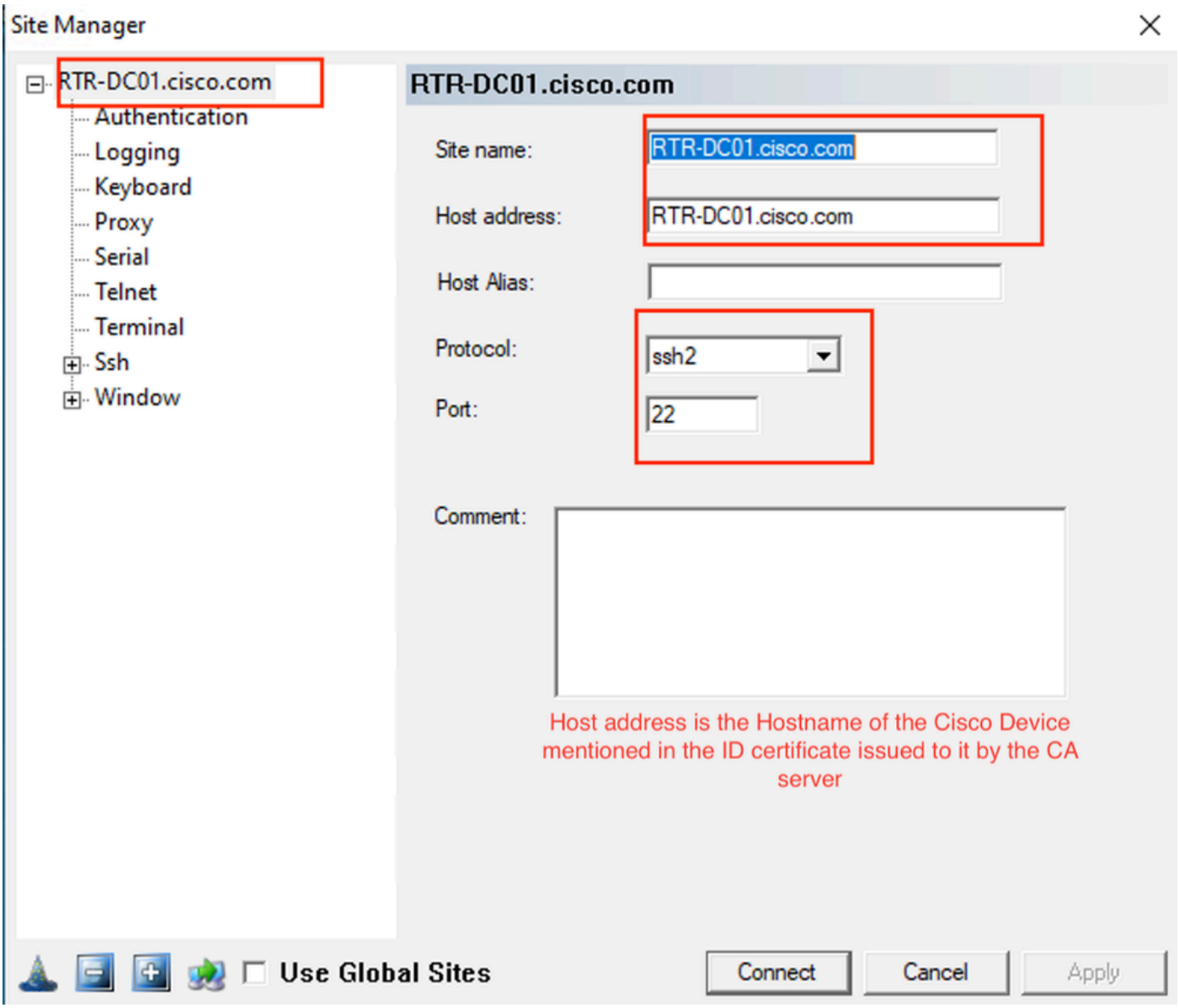


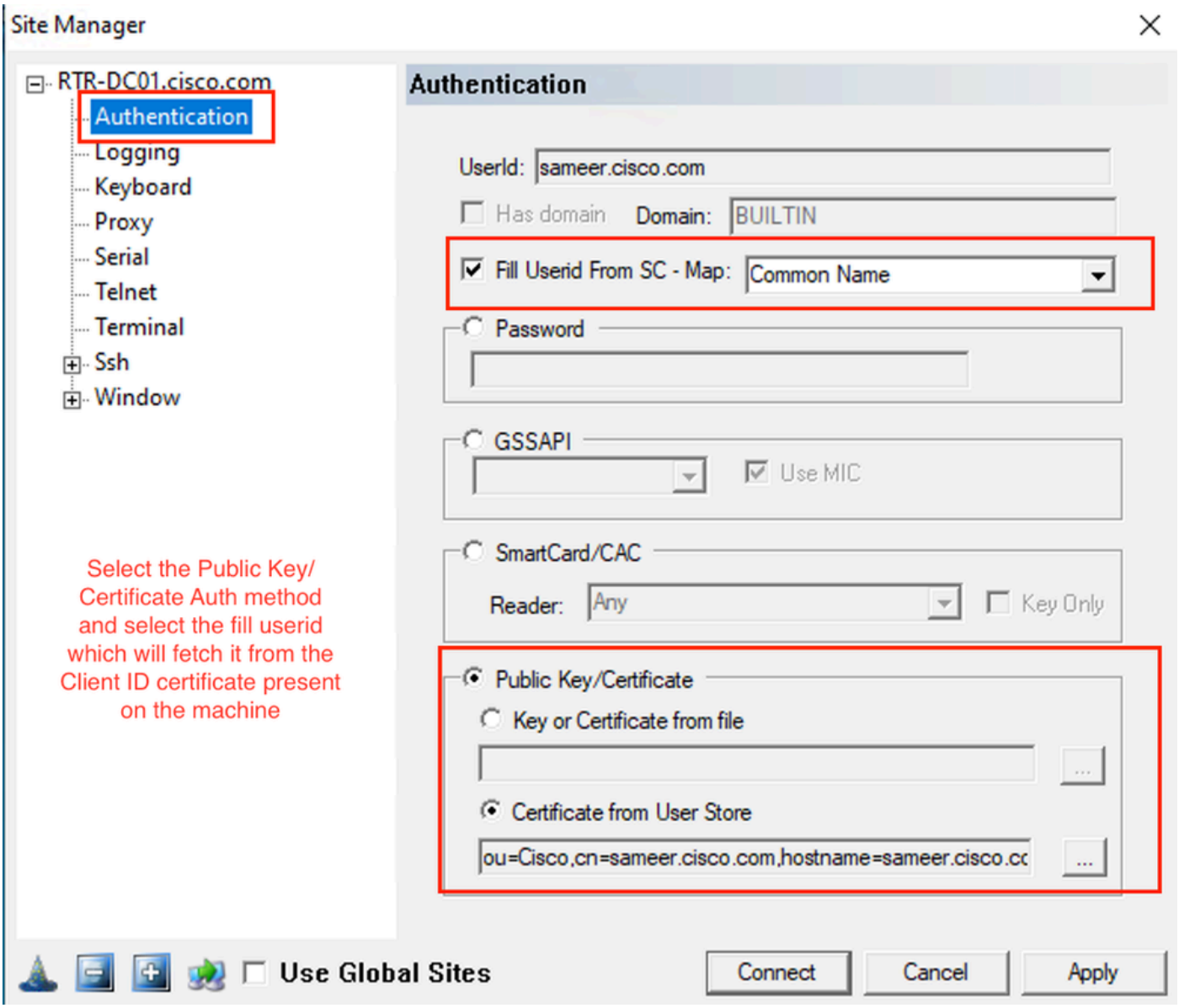
Remarque : Assurez-vous que le serveur SSH et le client SSH ont le certificat d'ID émis par le même serveur AC.



Remarque : Si un client SSH tel qu'une machine Windows possède un certificat d'ID émis par une autre autorité de certification, faites-le importer sur le serveur SSH, c'est-à-dire le commutateur Cisco et mappez ce point de confiance au profil de certificat de serveur SSH ci-dessus sous la section utilisateur.

Pragma Fortress CL (client SSH installé sur la machine utilisateur)


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



Vérifier

```
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
```

Dépannage

Ces débogages sont utilisés afin de suivre la session réussie :

```
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
```

Problèmes courants

Erreurs de configuration

La validation du certificat a réussi pour l'utilisateur. Cependant, en raison de l'absence de la commande obligatoire authorization username subjectname commonname dans la configuration, l'authentification du certificat pour l'utilisateur échoue.

```
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'
```

Informations connexes

- Guide de configuration PKI
- Assistance technique de Cisco et téléchargements

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.