Ridimensionamento delle chiavi SSH RSA predefinite sui bordi della scheda SD-WAN Cisco IOS XE

Sommario

Introduzione

In questo documento viene descritto come aumentare a una lunghezza maggiore le chiavi SSH RSA predefinite usate per i protocolli sicuri sui bordi Cisco IOS® XE SD-WAN.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Software Cisco Catalyst Defined Wide Area Network (SD-WAN)
- Chiavi SSH e funzionamento di base del certificato
- Algoritmo RSA

Componenti usati

- Cisco IOS® XE Catalyst SD-WAN Edge 17.9.4a

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Premesse

Secure Shell (SSH) è un protocollo di rete che consente agli utenti di stabilire connessioni remote ai dispositivi anche su una rete non protetta. Il protocollo protegge le sessioni utilizzando meccanismi di crittografia standard basati su un'architettura client-server.

RSA è Rivest, Shamir, Adleman: Algoritmo di crittografia (sistema di crittografia a chiave pubblica) che utilizza due chiavi: Chiave pubblica e privata, nota anche come coppia di chiavi. La chiave RSA pubblica è la chiave di crittografia, la chiave RSA privata è la chiave di decrittografia.

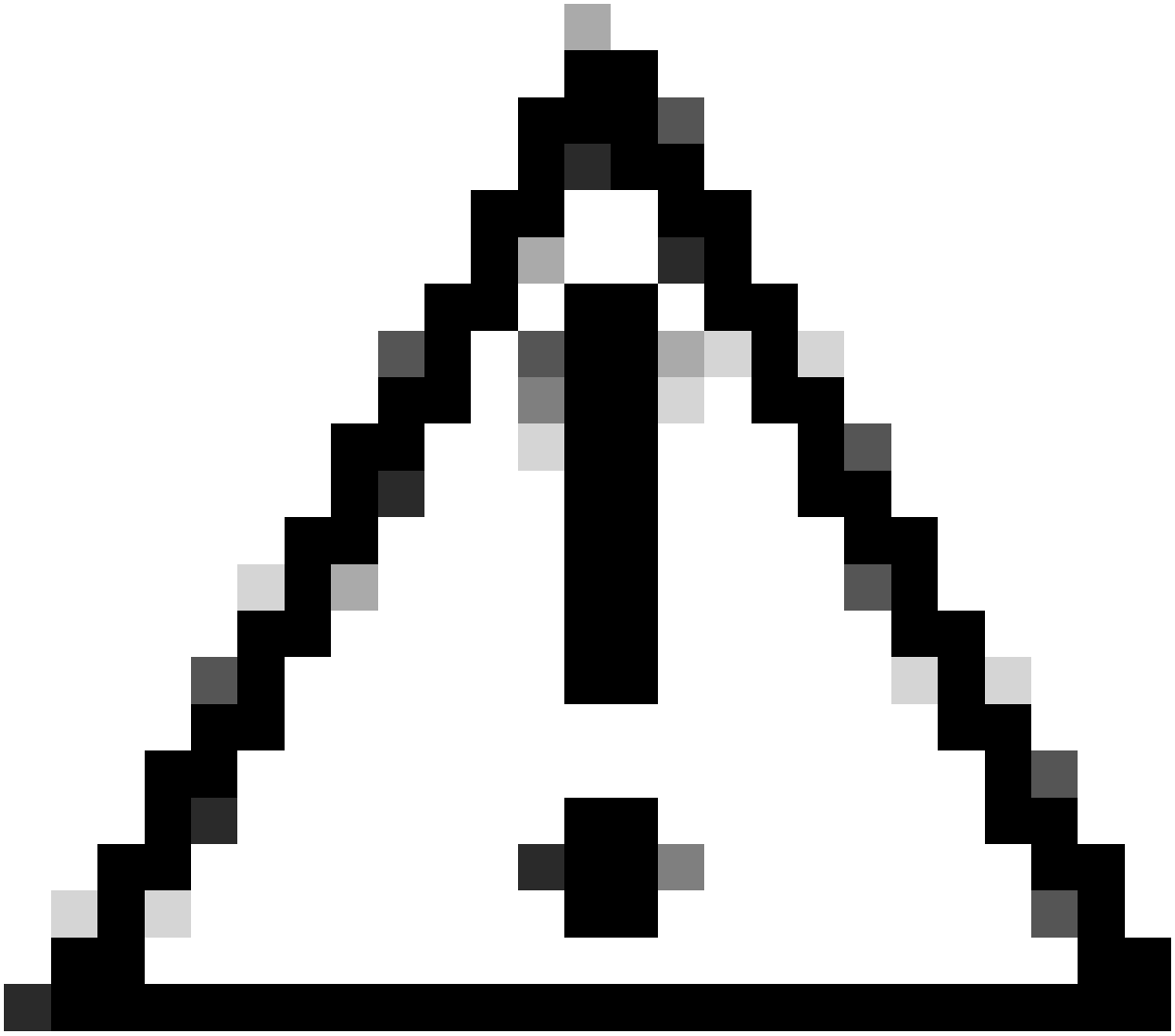
Le chiavi RSA hanno una lunghezza definita, in bit, del modulo. Quando si dice che una chiave RSA abbia una lunghezza di 2048 bit, in realtà il valore del modulo è compreso tra 22047 e 22048. Poiché le chiavi pubbliche e private di una determinata coppia condividono lo stesso modulo, per definizione hanno anche la stessa lunghezza.

Un certificato del trust point è un certificato autofirmato, da cui il nome del trust point, poiché non si basa sull'attendibilità di altri o di altre parti.

L'infrastruttura a chiave pubblica (PKI) di Cisco IOS consente di gestire i certificati per supportare protocolli di sicurezza quali IPSec (IP Security), SSH (Secure Shell) e SSL (Secure Sockets Layer).

Le chiavi SSH RSA sono importanti su Cisco Catalyst SD-WAN in quanto vengono utilizzate dal protocollo SSH per stabilire la comunicazione tra SD-WAN Manager e i dispositivi SD-WAN Edge. SD-WAN Manager utilizza il protocollo Netconf, che funziona su SSH per gestire, configurare e monitorare i dispositivi.

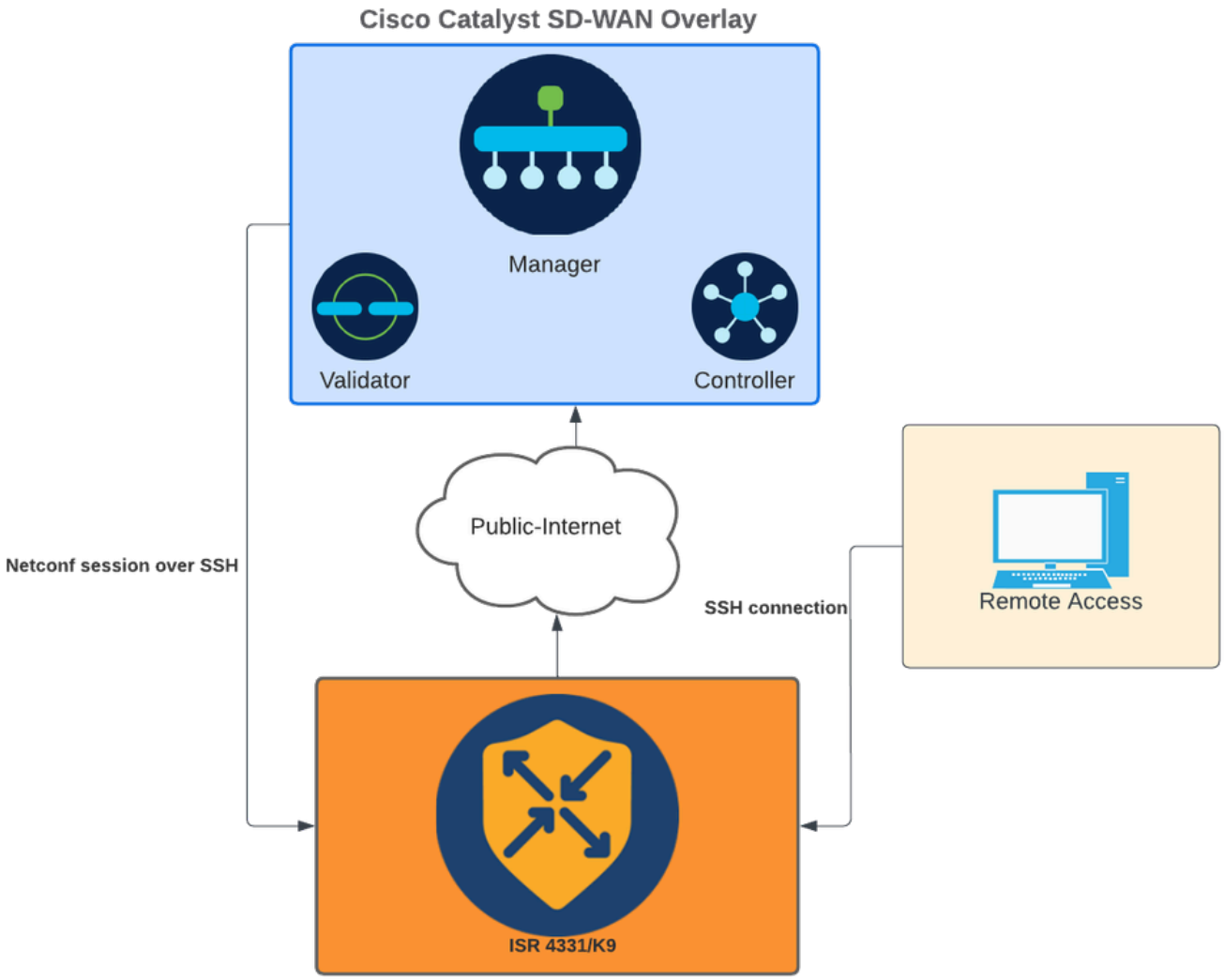
Per questo motivo, è necessario che le chiavi siano sempre sincronizzate e aggiornate. Se per conformità e controllo, è necessario modificare la lunghezza della chiave per la sicurezza, è necessario completare il processo descritto in questo documento per ridimensionare le chiavi e sincronizzarle correttamente sul certificato per evitare la disconnessione tra SD-WAN Manager e i dispositivi SD-WAN Edge.



Attenzione: Completare tutti i passaggi della procedura per evitare la perdita dell'accesso al dispositivo. Se il dispositivo è in produzione, si consiglia di eseguirlo in una finestra di manutenzione e di avere accesso da console al dispositivo.

Configurazione

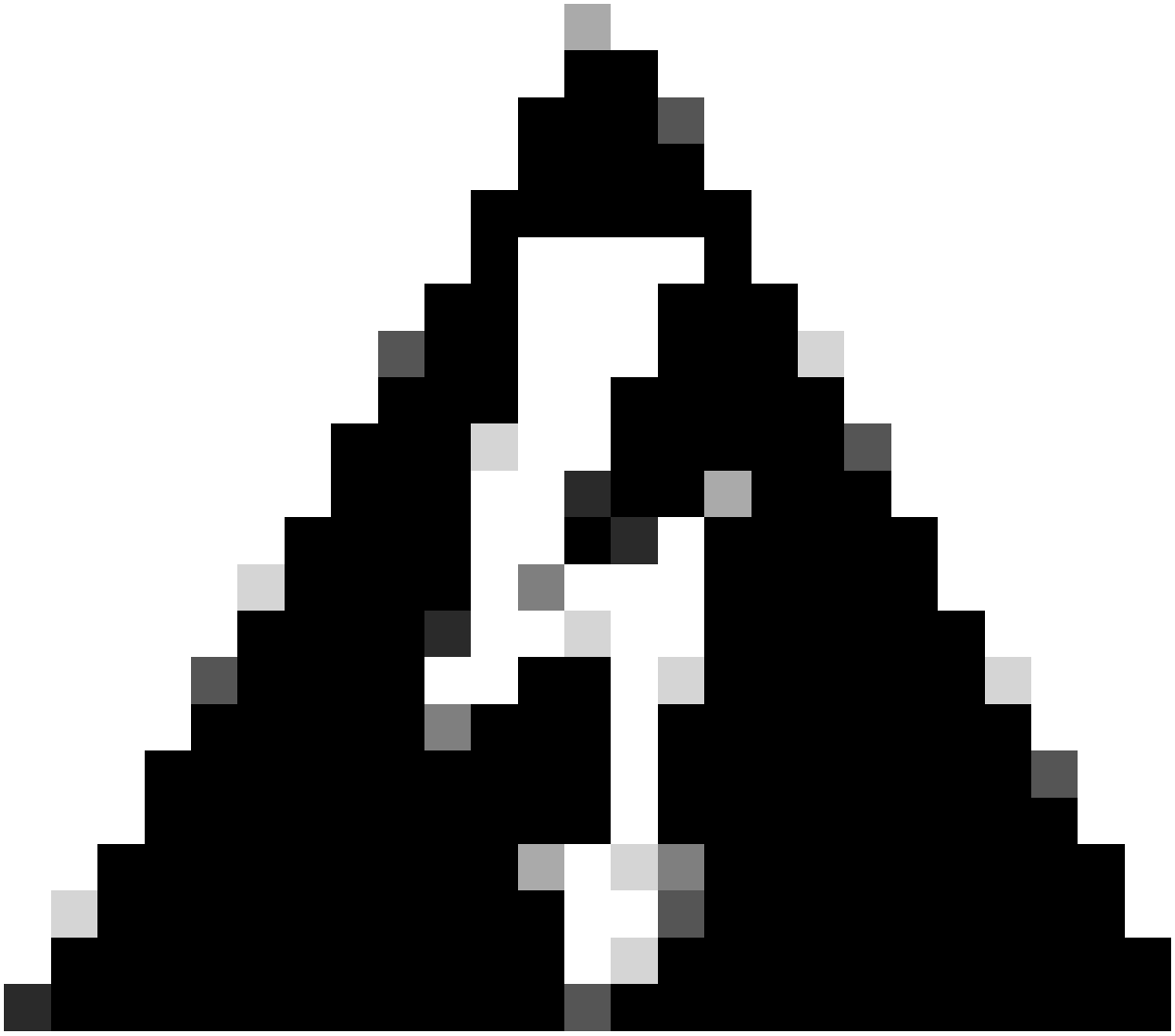
Esempio di rete



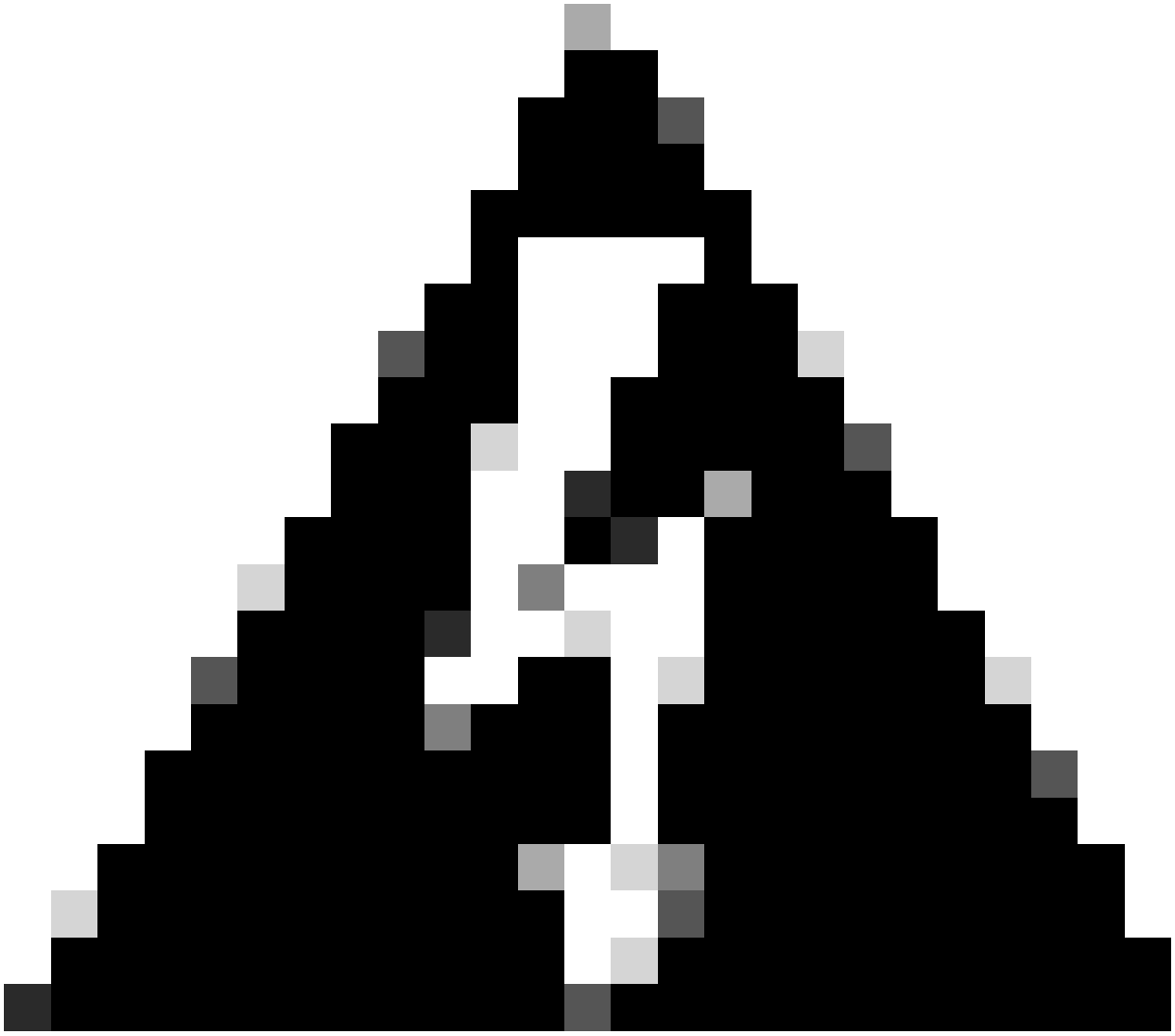
Esempio di rete

Configurazioni

le chiavi RSA nei dispositivi periferici della WAN possono essere modificate solo tramite l'interfaccia della riga di comando (CLI); Impossibile utilizzare i modelli di funzionalità aggiuntive CLI per aggiornare le chiavi.



Avviso: Si consiglia di eseguire il processo con la console poiché lo strumento SD-WAN Manager SSH non è disponibile fino al termine del processo.



Avviso: Questo processo richiede il riavvio del dispositivo. Se il dispositivo è in produzione, si consiglia di eseguirlo in una finestra di manutenzione e di avere accesso da console al dispositivo. Se l'accesso alla console è assente, configurare temporaneamente un altro protocollo di accesso remoto come telnet.

Nell'esempio di configurazione viene mostrato come rimuovere RSA 2048 e utilizzare la chiave RSA 4096.

1 - Ottenere il nome della chiave SSH corrente.

<#root>

```
Device#
```

```
show ip ssh
```

```
SSH Enabled - version 2.0
Authentication methods:publickey,keyboard-interactive,password
```

```
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecds
Hostkey Algorithms:x509v3-ssh-rsa,rsa-sha2-512,rsa-sha2-256,ssh-rsa
Encryption Algorithms:aes128-gcm,aes256-gcm,aes128-ctr,aes192-ctr,aes256-ctr
MAC Algorithms:hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,hmac-sha2-256,hmac-sha2-512,
KEX Algorithms:ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group14-sha1
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 2048 bits
IOS Keys in SECSH format(ssh-rsa, base64 encoded):
```

TP-self-signed-1072201169 <<<< RSA Key Name

Modulus Size : 2048 bits

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDZ5urq7f/X+AZJjUnMOdF9pLX+V0jPR8arK6bLSU7d
iGeSDDwW2MPNck/U5HBry9P/L4nKyZ1oevAhfy7cJVVmoHD41NQW9wb/hLtimuujnRRYkKuIWLmoI7AH
y6YQoetew8XVglVIjva+JzQ5ZX1JGm8AzN6a95RbRNhGRzgz9cTFmD7m6ArIKZPMyYQabXfrY+m/HuQ2
aytbHtJMgmOQk2fLPak03PnQNYXpiDP3CmOEh3LJg82FZQ1eohmhm+mAInwU4mlLHUouigyBuq1KEBVe
z3vxjB9X8rGF3qzUcx2lpHmhXaNPXWen2QQbyfAIDo8WXVoff24uLYlwCVkv
```

2 - Ottenere il certificato autofirmato del trust point corrente.

<#root>

```
Device#
```

show crypto pki trustpoint

Trustpoint TP-self-signed-1072201169: <<<< Self-signed Trustpoint name

```
Subject Name:
cn=IOS-Self-Signed-Certificate-1072201169
Serial Number (hex): 01
Persistent self-signed certificate trust point
Using key label
```

TP-self-signed-1072201169

Entrambi i nomi dei valori devono corrispondere.

3 - Eliminare la chiave corrente.

<#root>

```
Device#
```

crypto key zeroize rsa

4 - Verificare che la vecchia chiave sia stata eliminata correttamente.

<#root>

```
Device#
```

show ip ssh

5 - Generare la nuova chiave.

<#root>

```
Device#
```

crypto key generate rsa modulus 4096 label

```
The name for the keys will be: TP-self-signed-1072201169
% The key modulus size is 4096 bits
% Generating crypto RSA keys in background ...
*Jun 25 21:35:18.919: %CRYPTO_ENGINE-5-KEY_ADDITION: A key named TP-self-signed-1072201169 has been gen
*Jun 25 21:35:18.924: %SSH-5-ENABLED: SSH 2.0 has been enabled
*Jun 25 21:35:23.205: %CRYPTO_ENGINE-5-KEY_ADDITION: A key named TP-self-signed-1072201169 has been gen
*Jun 25 21:35:29.674: %SYS-6-PRIVCFG_ENCRYPT_SUCCESS: Successfully encrypted private config file
```

Il completamento di questo processo può richiedere da 2 a 5 minuti.

6 - Convalidare la nuova chiave generata.

<#root>

```
Device#
```

show ip ssh

```
SSH Enabled - version 2.0
Authentication methods:publickey,keyboard-interactive,password
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecds
Hostkey Algorithms:x509v3-ssh-rsa,rsa-sha2-512,rsa-sha2-256,ssh-rsa
Encryption Algorithms:aes128-gcm,aes256-gcm,aes128-ctr,aes192-ctr,aes256-ctr
MAC Algorithms:hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,hmac-sha2-256,hmac-sha2-512,
```

```
KEX Algorithms:ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group14-sha1
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 2048 bits
```

IOS Keys in SECSH format(ssh-rsa, base64 encoded): TP-self-signed-1072201169

Modulus Size : 4096 bits <<<< Key Size

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQDEOt/SX3oQKN6zOWvOaFAkMcaZNzQ6JgP+7xjuX143
YS7YGmOPwIPgs8N2LWvmdLXQ/PqsQOGGsdxo2+2Y/idAFm8O8mb6bcWFU+t3b/Pf6GBzUv8SPnR4i4nN
5GYhZE9HX3REWYp7d+7l1YawrDzpJ6d8RgUWLOtgHSzQ7P796cOB1YLtK3eFOOH1AFmFy5ec8Own7ikO
JjKtwEozImFMjHZfUEUjFuhPJELBO6yYEipPWMRaZYFfTRbNJm8/7SOJGlFkgFVW5nITTIgISoMV8EJv
bLl8cVgATDb1OckeDb7uU6PDXm3zonmZCOyqHtF1OAOJxUpUa6Iry1XwMzzZqDdu32F5If4/SSCmbHV2
46P8AjCdu/2TKK5etOO49UHOyObMgPuWrJpwtk1iYA3+t6N/QdlC5VSoua+TsMfp7Dh3k6qUTFUSy2h3
Kiibov1HKyvkcqXi6nDfAKb8o+Z8/43xbvWlDIKAuj1rbdyqPAJB4llTZJkOHk8zRP5gZ8u4jTjNKQHb
vNa3ieg4RLEDOx4lqCk+iSRzdddMq2te1xSWFPh67i4BnJHvhVnR6LF5Gu+uF5TWwcpy2MMOul4YDJYr
D+jnyoZr4PnfWAgk4M9U89deWSlIRPMIXYd35YmLvD6OeQ5EQALNiNPUEkpdPKs4orYysEVOpRoY+HQ
```

Ora viene generata una nuova chiave. Tuttavia, al momento dell'eliminazione della vecchia chiave, anche il certificato autofirmato utilizzato dalle sessioni Netconf viene eliminato dal trust point.

<#root>

```
Device#
```

sh crypto pki trustpoint status

```
Trustpoint TP-self-signed-1072201169:
Issuing CA certificate configured::
Issuing CA certificate configured:
Subject Name:
cn=Cisco Licensing Root CA,o=Cisco
Fingerprint MD5: 1468DC18 250BDFCF 769C29DF E1F7E5A8
Fingerprint SHA1: 5CA95FB6 E2980EC1 5AFB681B BB7E62B5 AD3FA8B8
State:
```

Keys generated ..... No <<<< Depending on the version, it can erase the key or even that, delete

```
Issuing CA authenticated ..... Yes
Certificate request(s) ..... None
```

Dopo aver generato la nuova chiave 4096, le chiavi non vengono aggiornate automaticamente nel certificato autofirmato ed è necessario completare ulteriori passaggi per aggiornarla.

Nota: Se solo la chiave viene generata, ma non viene aggiornata nel certificato, SD-WAN Manager perde le sessioni Netconf e ciò potrebbe interrompere tutte le attività di gestione verso il dispositivo (modelli, configurazione, e così via).

È possibile generare il certificato o assegnare la chiave in due modi:

1 - Riavviare il dispositivo.

```
<#root>

Device#

reload
```

2 - Riavviare HTTP secure-server. Questa opzione è disponibile solo se il dispositivo è in modalità CLI.

```
<#root>

Device (config)#

no ip http secure-server

Device (config)#

commit

Device (config)#

ip http secure-server

Device (config)#

commit
```

Verifica

Dopo il ricaricamento, verificare che la nuova chiave sia stata generata e che il certificato si trovi in un trust point con lo stesso nome.

```
<#root>

Device#

 show ip ssh

SSH Enabled - version 2.0
Authentication methods:publickey,keyboard-interactive,password
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecds
Hostkey Algorithms:x509v3-ssh-rsa,rsa-sha2-512,rsa-sha2-256,ssh-rsa
Encryption Algorithms:aes128-gcm,aes256-gcm,aes128-ctr,aes192-ctr,aes256-ctr
MAC Algorithms:hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,hmac-sha2-256,hmac-sha2-512,
KEX Algorithms:ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group14-sha1
```

```
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 2048 bits
```

IOS Keys in SECSH format(ssh-rsa, base64 encoded): TP-self-signed-1072201169

Modulus Size : 4096 bits

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQDEOt/SX3oQKN6zOWvOaFAkMcaZNzQ6JgP+7xjuX143
YS7YGmOPwIPgs8N2LWvmdLXQ/PqsQOGGsdxo2+2Y/idAFm8O8mb6bcWFU+t3b/Pf6GBzUv8SPnR4i4nN
5GYhZE9HX3REWYp7d+7l1YawrDzpJ6d8RgUWLOtgHSzQ7P796c0B1YLtK3eFOOH1AFmFy5ec8Own7ikO
JjKtwEozImFMjHZfUEUjFuhPJELBO6yYEipPWMRaZYFfTRbNJm8/7SOJGlFkgFVW5nITTIgISoMV8EJv
bLl8cVgATDb1OckeDb7uU6PDXm3zonmZCOyqHtF1OAOJxUpUa6Iry1XwMzzZqDdu32F5If4/SSCmbHV2
46P8AjCdu/2TKK5etOO49UHOyObMgPuWrJpwtk1iYA3+t6N/QdlC5VSoua+TsMfp7Dh3k6qUTFUSy2h3
Kiibov1HKyvkcqXi6nDfAKb8o+Z8/43xbvWlDIKAuj1rbdyqPAJB4llTZJkOHk8zRP5gZ8u4jTjNKQHb
vNa3ieg4RLEDOx4lqCk+iSRzdddMq2te1xSWFPh67i4BnJHvhVnR6LF5Gu+uF5TWwcpy2MMOul4YDJYr
D+jnyoZr4PnfWAgk4M9U89deWSlIRPMIXYd35YmLvD6OeQ5EQALNiNPUEkpdPKs4orYysEVOpRoY+HQ
```

<#root>

```
Device#
```

show crypto pki trustpoint

Trustpoint TP-self-signed-1072201169: <<<< Trustpoint name

```
Subject Name:
cn=IOS-Self-Signed-Certificate-1072201169
Serial Number (hex): 01
Persistent self-signed certificate trust point
```

Using key label TP-self-signed-107220116

<#root>

```
Device#
```

show crypto pki certificates

```
Router Self-Signed Certificate
Status: Available
Certificate Serial Number (hex): 01
Certificate Usage: General Purpose
Issuer:
cn=IOS-Self-Signed-Certificate-1072201169
Subject:
Name: IOS-Self-Signed-Certificate-1072201169
cn=IOS-Self-Signed-Certificate-1072201169
Validity Date:
start date: 21:07:33 UTC Dec 27 2023
end date: 21:07:33 UTC Dec 26 2033
```

Associated Trustpoints: TP-self-signed-1072201169

```
Storage: nvram:IOS-Self-Sig#4.cer
```

Confermare che SD-WAN Manager possa applicare le modifiche alla configurazione al router del dispositivo.

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l’accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).