Cisco IOS XE SD-WAN 에지에서 기본 SSH RSA 키 크기 조정

목차

소개

이 문서에서는 보안 프로토콜에 사용되는 기본 SSH RSA 키를 Cisco IOS® XE SD-WAN Edge에서 더 긴 길이로 늘리는 방법에 대해 설명합니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- Cisco Catalyst SD-WAN(Software-Defined Wide Area Network)
- SSH 키 및 인증서 기본 작업
- RSA 알고리즘

사용되는 구성 요소

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

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

배경 정보

SSH(Secure Shell)는 사용자가 보호되지 않은 네트워크를 통해서도 장치에 대한 원격 연결을 설정할 수 있는 네트워크 프로토콜입니다. 이 프로토콜은 클라이언트-서버 아키텍처에 기반한 표준 암

호화 메커니즘을 사용하여 세션을 보호합니다.

RSA는 Rivest, Shamir, Adleman입니다. 두 개의 키를 사용하는 암호화 알고리즘(공개 키 암호화 시스템): 공개 및 개인 키(키 쌍이라고도 함) 공용 RSA 키는 암호화 키이고 개인 RSA 키는 암호 해독 키입니다.

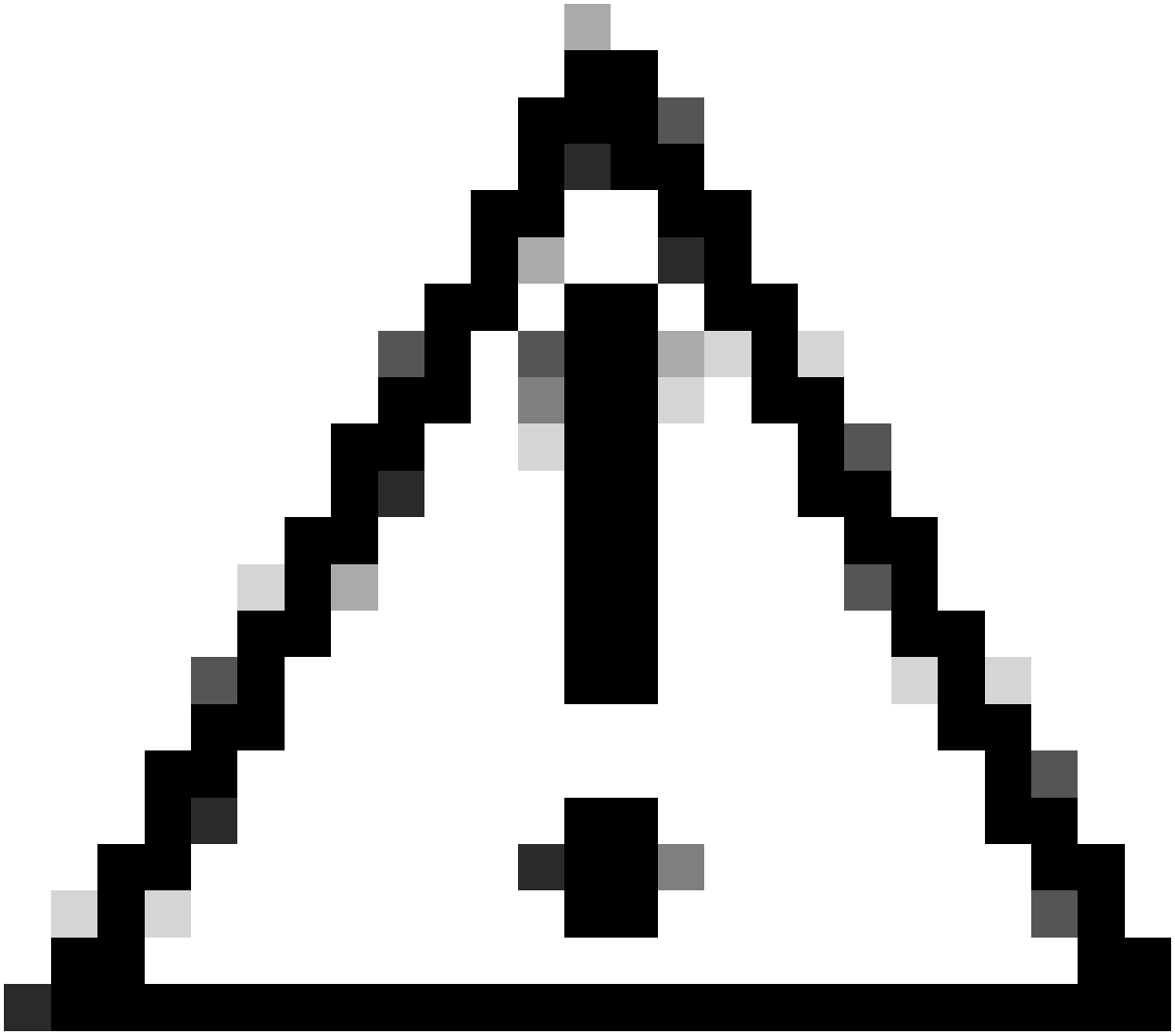
RSA 키는 모듈러스의 정의된 길이(비트)를 가집니다. RSA 키의 길이를 2048비트라고 하면 모듈러스 값이 22047과 22048 사이에 있다는 것을 의미합니다. 특정 쌍의 공개 키와 개인 키는 동일한 모듈러스를 공유하므로 정의상 동일한 길이를 갖습니다.

신뢰 지점 인증서는 자체 서명 인증서이므로 다른 사용자 또는 다른 사람의 신뢰에 의존하지 않으므로 신뢰 지점이라는 이름을 갖습니다.

Cisco IOS PKI(Public Key Infrastructure)는 IPSec(IP Security), SSH(Secure Shell), SSL(Secure Socket Layer)과 같은 보안 프로토콜을 지원하는 인증서 관리 기능을 제공합니다.

SSH RSA 키는 SSH 프로토콜에서 SD-WAN Manager와 SD-WAN Edge 디바이스 간의 통신을 설정하는 데 사용되므로 Cisco Catalyst SD-WAN에서 중요합니다. SD-WAN Manager는 SSH를 통해 작동하는 Netconf 프로토콜을 사용하여 디바이스를 관리, 구성 및 모니터링하기 때문입니다.

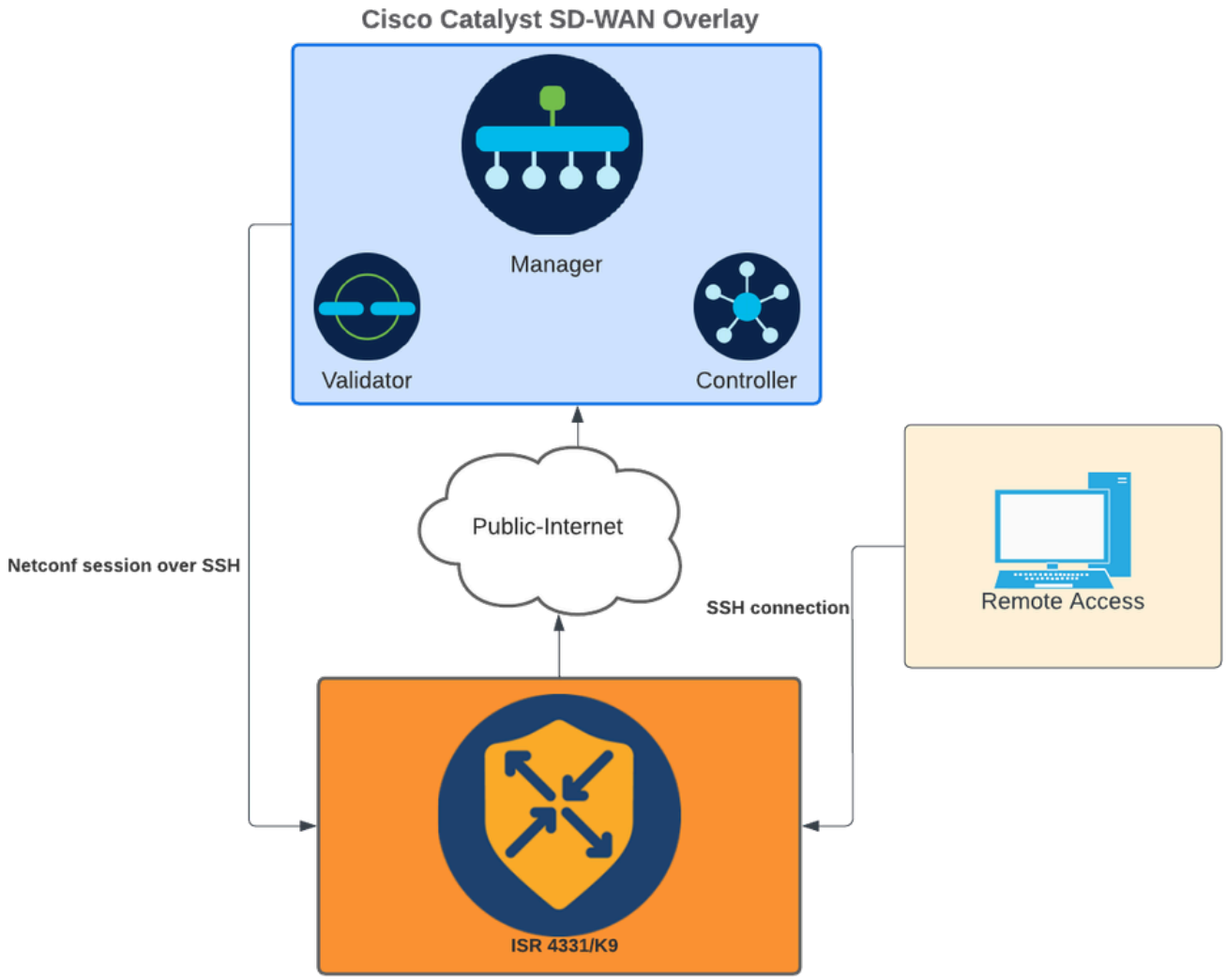
따라서 항상 키가 동기화되고 업데이트되어야 합니다. 규정 준수 및 감사를 통해 보안을 위해 키 길이를 수정해야 하는 경우, SD-WAN Manager와 SD-WAN Edge 디바이스 간의 연결을 방지하기 위해 키의 크기를 조정하는 프로세스를 완료하고 인증서에서 올바르게 동기화해야 합니다.



주의: 디바이스에 대한 액세스 권한이 손실되지 않도록 프로세스의 모든 단계를 완료하십시오. 디바이스가 프로덕션 상태인 경우 유지 관리 창에서 수행하고 디바이스에 대한 콘솔 액세스 권한을 갖는 것이 좋습니다.

구성

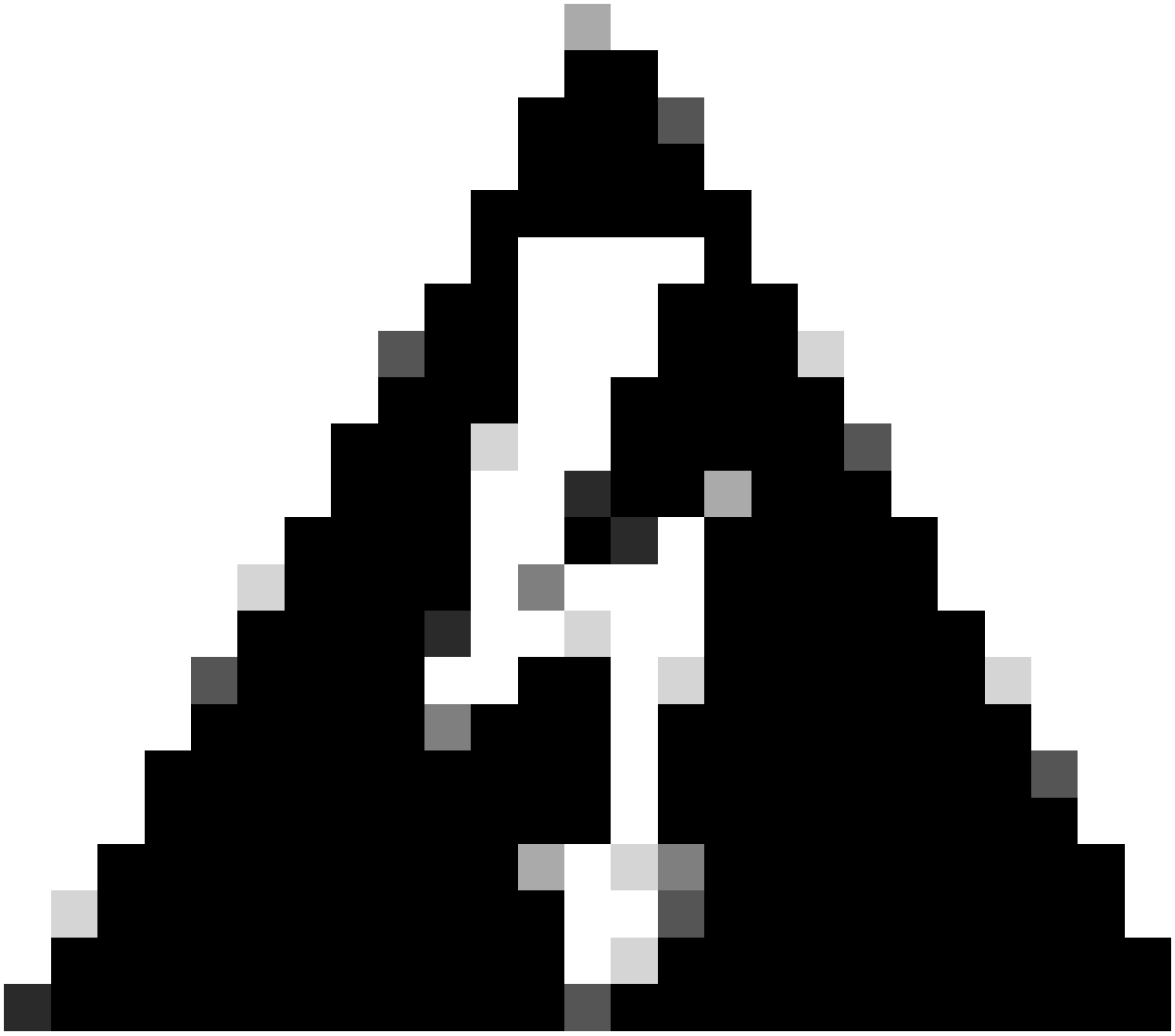
네트워크 다이어그램



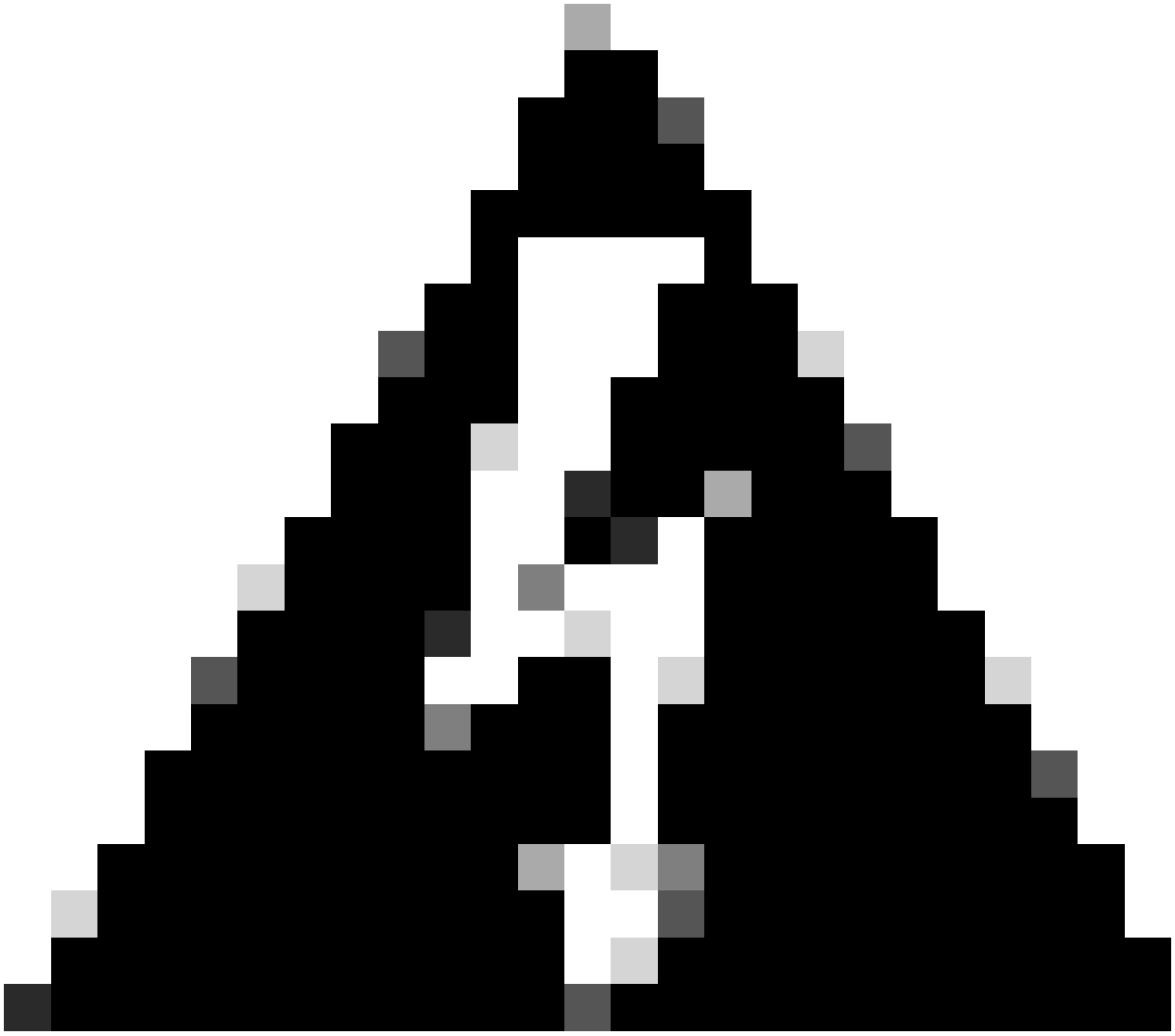
네트워크 다이어그램

설정

WAN 에지 디바이스의 RSA 키는 CLI(Command Line Interface)를 통해서만 수정할 수 있습니다. CLI 애드온 기능 템플릿은 키를 업데이트하는 데 사용할 수 없습니다.



경고: 프로세스가 완료될 때까지 SD-WAN Manager SSH Tool을 사용할 수 없으므로 콘솔을 사용하여 프로세스를 수행하는 것이 좋습니다.



경고: 이 프로세스에는 디바이스를 다시 시작해야 합니다. 디바이스가 프로덕션 상태인 경우 유지 관리 창에서 수행하고 디바이스에 대한 콘솔 액세스 권한을 갖는 것이 좋습니다. 콘솔 액세스가 없는 경우 임시로 다른 원격 액세스 프로토콜을 텔넷으로 구성합니다.

이 컨피그레이션 예에서는 RSA 2048을 제거하고 RSA 4096 키를 사용하는 방법을 보여줍니다.

1 - 현재 SSH 키 이름을 가져옵니다.

<#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+VOjPR8arK6bLSU7d
iGeSDDwW2MPNck/U5HBry9P/L4nKyZ1oevAhfy7cJVVmoHD41NQW9wb/hLtimuujnRRYkKuIWLmoI7AH
y6YQoetew8XVglVIjva+JzQ5ZX1JGm8AzN6a95RbRNhGRzgz9cTFmD7m6ArIKZPMyYQabXfrY+m/HuQ2
aytbHtJMgmOQk2fLPakO3PnQNYXpiDP3CmOEh3LJg82FZQ1eohmhm+mAInwU4mlLHUouigyBuq1KEBVe
z3vxjB9X8rGF3qzUcx2lpHmhXaNPXWen2QQbyfAIDo8WXVoff24uLYlwCVkv
```

2 - 현재 신뢰 지점 자체 서명 인증서를 가져옵니다.

<#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

두 value-name이 모두 일치해야 합니다.

3 - 현재 키를 삭제합니다.

<#root>

```
Device#
```

crypto key zeroize rsa

4 - 이전 키가 성공적으로 삭제되었는지 확인합니다.

<#root>

Device#

show ip ssh

5 - 새 키를 생성합니다.

<#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
```

이 프로세스는 완료하는 데 2분에서 5분 정도 걸릴 수 있습니다.

6 - 생성된 새 키를 확인합니다.

<#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/SX3oQKN6z0WvOaFAkMcaZNzQ6JgP+7xjuX143
YS7YGm0PwIPgs8N2LWvmdLXQ/PqsQ0GGsdxo2+2Y/idAFm8O8mb6bcWFU+t3b/Pf6GBzUv8SPnR4i4nN
5GYhZE9HX3REWYp7d+7l1YawrDzpJ6d8RgUWLOtgHSzQ7P796c0B1YLtK3eFO0H1AFmFy5ec8Own7ik0
JjKtwEozImFMjHZfUEUjFuhPJELBO6yYEipPWMRaZYFfTRbNJm8/7SOJGlFkgFVW5nITTIgISoMV8EJv
bLl8cVgATDb1OckeDb7uU6PDXm3zonmZC0yqHtF1OAOJxUpUa6Iry1XwMzzZqDdu32F5If4/SSCmbHV2
46P8AjCdu/2TKK5et0O49UH0y0bMgPuWrJpwtk1iYA3+t6N/QdlC5VSoua+TsMfp7Dh3k6qUTFUSy2h3
Kiibov1HKyvkcqXi6nDfAKb8o+Z8/43xbvWlDIKAuj1rbdyqPAJB4llTZJkOHk8zRP5gZ8u4jTjNKQHb
vNa3ieg4RLEDOx4lqCk+iSRzdddMq2te1xSWFPh67i4BnJHvhVnR6LF5Gu+uF5TWwcpy2MMOul4YDJYr
D+jnyoZr4PnfWAgk4M9U89deWSlIRPMIXYd35YmLvD6OeQ5EQALNiNPUEkpdPKs4orYysEV0pRoY+HQ
```

이제 새 키가 생성됩니다. 그러나 이전 키가 삭제되는 순간 Netconf 세션에서 사용 중인 자체 서명 인증서도 신뢰 지점에서 삭제됩니다.

```
<#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
```

새 4096 키가 생성되면 자체 서명 인증서에서 키가 자동으로 업데이트되지 않으며 이를 업데이트하기 위해 추가 단계를 완료해야 합니다.

참고: 키만 생성되지만 인증서에서 업데이트되지 않으면 SD-WAN Manager는 Netconf 세션을 잃게 되며, 이는 디바이스에 대한 모든 관리 활동(템플릿, 컨피그레이션 등)을 중단시킬 수 있습니다.

두 가지 방법으로 인증서를 생성하고 키를 할당할 수 있습니다.

1 - 디바이스를 다시 로드합니다.

<#root>

Device#

reload

2 - HTTP 보안 서버를 다시 시작합니다. 이 옵션은 디바이스가 CLI 모드에 있는 경우에만 사용할 수 있습니다.

<#root>

Device (config)#

no ip http secure-server

Device (config)#

commit

Device (config)#

ip http secure-server

Device (config)#

commit

다음을 확인합니다.

다시 로드한 후 새 키가 생성되고 인증서가 같은 이름의 신뢰 지점에 있는지 확인합니다.

<#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/SX3oQKN6z0WvOaFAkMcaZNzQ6JgP+7xjuX143
YS7YGm0PwIPgs8N2LWvmdLXQ/PqsQ0GGsdxo2+2Y/idAFm8O8mb6bcWFU+t3b/Pf6GBzUv8SPnR4i4nN
5GYhZE9HX3REWYp7d+7l1YawrDzpJ6d8RgUWLOtgHSzQ7P796c0B1YLtK3eFOOH1AFmFy5ec8Own7ik0
JjKtwEozImFMjHZfUEUjFuhPJELBO6yYEipPWMRaZYFfTRbNJm8/7SOJGlFkgFVW5nITTIgISoMV8EJv
bLl8cVgATDb1OckeDb7uU6PDXm3zonmZC0yqHtF1OAOJxUpUa6Iry1XwMzzZqDdu32F5If4/SSCmbHV2
46P8AjCdu/2TKK5etOO49UH0y0bMgPuWrJpwtk1iYA3+t6N/QdlC5VSoua+TsMfp7Dh3k6qUTFUSy2h3
Kiibov1HKyvkcqXi6nDfAKb8o+Z8/43xbvWlDIKAuj1rbdyqPAJB4llTZJkOHk8zRP5gZ8u4jTjNKQHb
vNa3ieg4RLEDOx4lqCk+iSRzdddMq2te1xSWFPh67i4BnJHvhVnR6LF5Gu+uF5TWwcpy2MMOul4YDJYr
D+jnyoZr4PnfWAgk4M9U89deWSlIRPMIXYd35YmLvD6OeQ5EQALNiNPUEkpdPKs4orYysEV0pRoY+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
```

SD-WAN Manager가 컨피그레이션 변경 사항을 디바이스 라우터에 적용할 수 있는지 확인합니다.

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.