Catalyst 9800 WLC에서 CSR 인증서 생성 및 다운로드

목차

소개

이 문서에서는 Catalyst 9800에서 인증서를 생성, 다운로드 및 설치하기 위한 전반적인 프로세스에 대해 설명합니다

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- 기본 작동을 위해 9800 WLC, 액세스 포인트(AP)를 구성하는 방법
- OpenSSL 애플리케이션 사용 방법
- PKI(Public Key Infrastructure) 및 디지털 인증서

사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- 9800-L, Cisco IOS® XE 버전 17.3.3
- OpenSSL 애플리케이션

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.

구성

16.10.X에서 9800s는 웹 인증 및 웹 관리를 위해 다른 인증서를 지원하지 않습니다. 포털의 웹 로그에서는 항상 기본 인증서를 사용합니다.

16.11.X에서는 웹 인증을 위한 전용 인증서를 구성하고 전역 매개변수 맵 내에서 신뢰 지점을 정의할 수 있습니다.

9800 WLC에 대한 인증서를 가져오는 두 가지 옵션이 있습니다.

1. OpenSSL 또는 기타 SSL 애플리케이션을 사용하여 CSR(Certificate Signing Request)을 생성합니다. CA(Certificate Authority)에서 서명한 PKCS12 인증서를 가져와 9800 WLC에 직접 로드합니다. 즉, 개인 키가 해당 인증서와 함께 번들로 제공됩니다.
2. 9800 WLC CLI를 사용하여 다음을 생성합니다. CSR은 CA에서 서명한 다음 체인의 각 인증서를 수동으로 9800 WLC에 로드합니다.

당신의 요구에 가장 적합한 것을 사용하세요.

옵션 1 - 기존 PKCS12 서명 인증서 로드

서명 요청 정의

아직 인증서가 없는 경우 CA에 제공할 서명 요청을 생성해야 합니다.

OpenSSL이 설치된 노트북 컴퓨터의 현재 디렉터리에서 openssl**.cnf 파일**을 편집하고 이 행을 복사하여 붙여 넣어 새로 생성된 CSR에 SAN(Subject Alternate Names) 필드를 포함합니다.

```
[ req ]
default_bits       = 4096
distinguished_name = req_distinguished_name
req_extensions     = req_ext
[ req_distinguished_name ]
countryName                 = Country Name (2 letter code)
stateOrProvinceName         = State or Province Name (full name)
localityName                = Locality Name (eg, city)
organizationName            = Organization Name (eg, company)
commonName                  = Common Name (e.g. server FQDN or YOUR name)
[ req_ext ]
subjectAltName = @alt_names
[alt_names]
DNS.1   = testdomain.com
DNS.2   = example.com
DNS.3   = webadmin.com
```

DNS.X 이름을 SAN으로 바꿉니다. 기본 필드를 필요한 인증서 세부 정보로 대체합니다. SAN 필드

(DNS.x) 내에서 Common Name(일반 이름)을 반복해야 합니다. Google Chrome에서 인증서를 신뢰하려면 URL에 있는 이름이 SAN 필드에 있어야 합니다.

웹 관리자의 경우, 브라우저 주소 표시줄의 URL에서 관리자가 어떤 유형을 사용하든 인증서가 일치하도록 SAN 필드에 URL의 변형(예: 호스트 이름만 또는 전체 FQDN(Fully Qualified Domain Name))을 채워야 합니다.

다음 명령을 사용하여 OpenSSL에서 CSR을 생성합니다.

```
openssl req -out myCSR.csr -newkey rsa:4096 -nodes -keyout private.key -config openssl.cnf
```

CSR은 명령에 전체 경로**가 제공되지** 않는 한 OpenSSL이 실행되는 디렉토리에서 myCSR**.csr**로, 해당 키를 private**.key**로 생성합니다.

private.key 파일은 통신을 암호화하는 데 사용되므로 안전하게 유지해야 합니다.

다음 항목을 사용하여 내용을 확인할 수 있습니다.

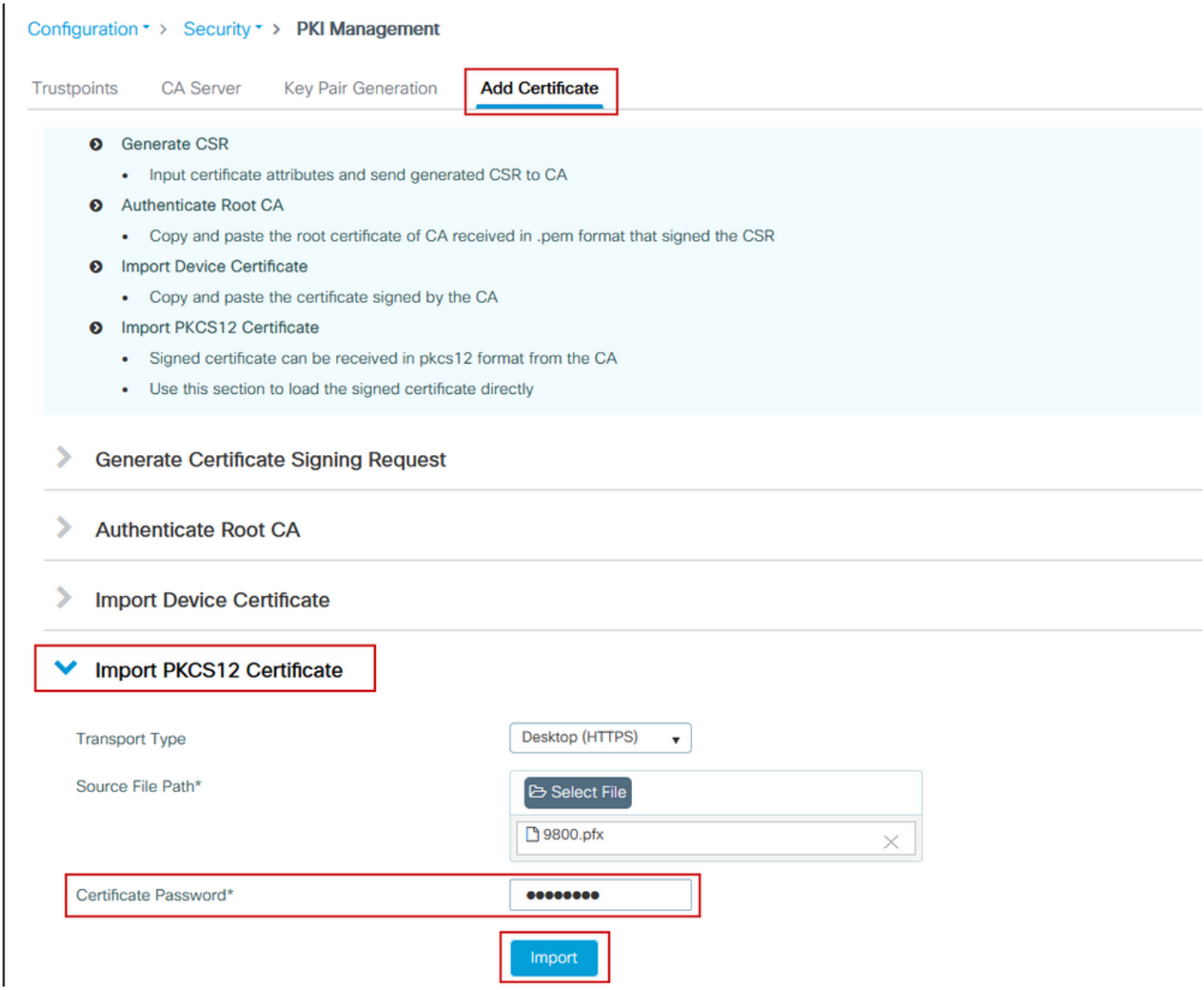
```
openssl req -noout -text -in myCSR.csr
```

그런 다음 이 CSR을 CA에 제공하여 서명하고 인증서를 받을 수 있습니다. 전체 체인이 CA에서 다운로드되고 추가 조작이 필요한 경우 인증서가 Base64 형식인지 확인합니다.

인증서 가져오기

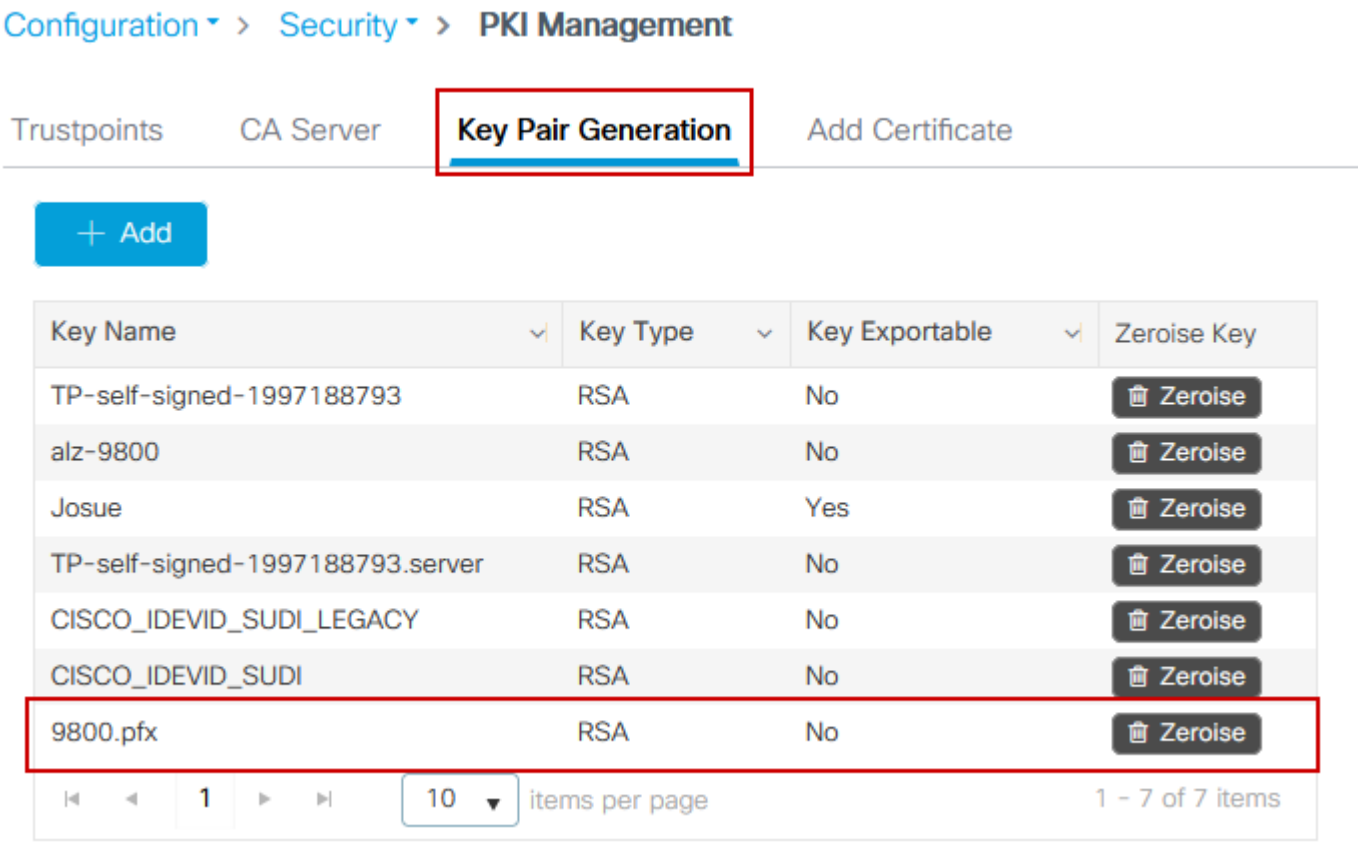
1단계. 9800 WLC에서 연결할 수 있는 TFTP(Trivial File Transfer Protocol) 서버에 PKCS12 인증서를 저장합니다. PKCS12 인증서는 개인 키 및 루트 CA까지의 인증서 체인을 포함해야 합니다.

2단계. 9800 WLC GUI를 열고 Configuration(컨피그레이션) **> Security(보안) > PKI Management(PKI 관리)로** 이동하고 **Add Certificate(인증서 추가) 탭**을 클릭합니다**.** Import PKCS**12 Certificate(PKCS12 인증서 가져오기)** 메뉴를 확장하고 TFTP 세부 정보를 입력합니다. 또는 Transport **Type** 드롭다운 **목록**의 Desktop**(HTTPS)** 옵션을 사용하면 브라우저를 통해 HTTP 업로드가 허용됩니다. **인증서 비밀번호**는 PKCS12 인증서가 생성될 때 사용된 비밀번호를 나타냅니다.



3단계. 정보가 올바른지 확인하고 Import(가져오기)를 **클릭합니다.** 그런 다음 Key Pair Generation(키 쌍 생성) 탭에 설치된 이 새 신뢰 지점에 대한 새 인증서 **키 쌍이** 표시됩니다. 가져오기에 성공하면 9800 WLC는 다중 레벨 CA에 대한 추가 신뢰 지점도 생성합니다.

> **참고**: 현재 9800 WLC는 특정 신뢰 지점이 webauth 또는 webadmin에 사용될 때마다 전체 인증서 체인을 제공하지 않고 디바이스 인증서 및 즉시 발급자를 제공합니다. 이는 Cisco 버그 ID CSCwa로 추적됩니다23606 Cisco IOS® XE 17.8에서 수정되었습니다.



CLI:

```
9800# configure terminal
9800(config)#crypto pki import
```

참고: 다중 레벨 CA에 대한 추가 신뢰 지점을 생성하려면 9800 WLC에 대해 인증서 파일 이름과 신뢰 지점 이름이 정확하게 일치해야 합니다.

다중 레벨 CA 시나리오에서 PKCS12 형식 변환 및 인증서 체인

PEM 또는 CRT 형식의 개인 키 파일과 인증서가 있으며 이를 PKCS12(.pfx) 형식으로 결합하여 9800 WLC에 업로드하려는 상황이 발생할 수 있습니다. 이렇게 하려면 다음 명령을 입력합니다.

```
openssl pkcs12 -export -in
```

인증서 체인(하나 이상의 중간 CA 및 루트 CA)이 모두 PEM 형식인 경우 모두 단일 .pfx 파일로 결합해야 합니다.

먼저 수동으로 단일 파일에 CA 인증서를 결합합니다. 내용을 복사하여 붙여 넣습니다(.pem 형식으

로 파일 저장).

```
----- BEGIN Certificate --------
<intermediate CA cert>
------END Certificate --------
-----BEGIN Certificate -----
<root CA cert>
-----END Certificate--------
```

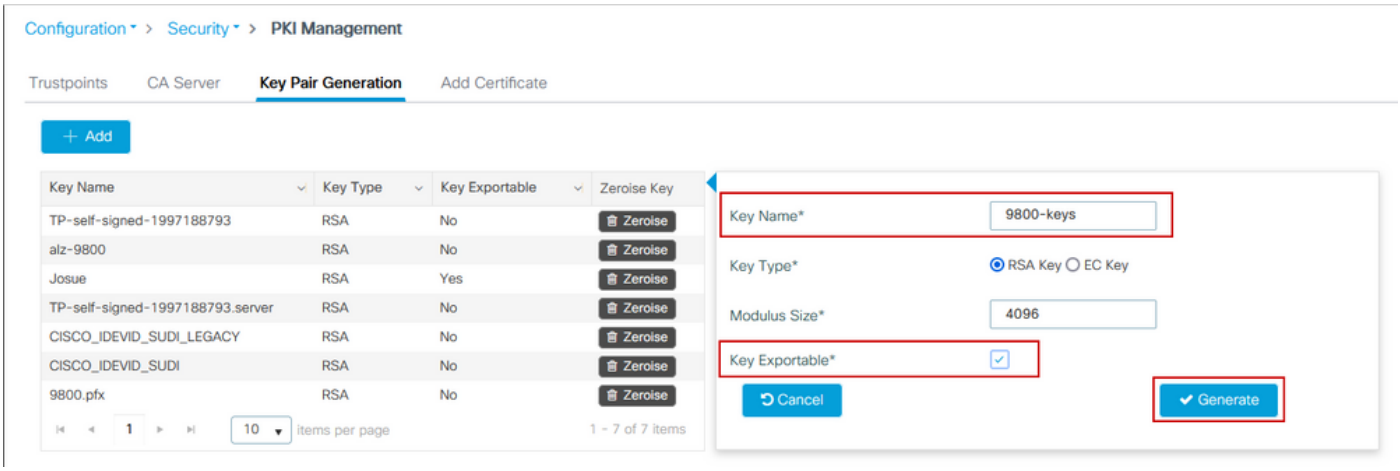
그런 다음 하나의 PKCS12 인증서 파일에 다음을 사용하여 모두 결합할 수 있습니다.

```
openssl pkcs12 -export -out chaincert.pfx -inkey
```

최종 인증서의 모양을 보려면 문서 끝의 Verify 섹션을 참조하십시오.

옵션 2 - 9800 WLC에서 키 및 서명 요청 정의

1단계. 범용 RSA 키 쌍을 생성합니다. Configuration(컨피그레이션) **> Security(보안) > PKI Management(PKI 관리)로 이동하고 Key Pair Generation(키 쌍 생성)** 탭을 선택한 다음 **+ Add(추가)를 클릭합니다**. 세부 정보를 입력하고 내보낼 수 있는 **키 확인란**이 선택되었는지 확인한 다음 생성을 **클릭합니다**.



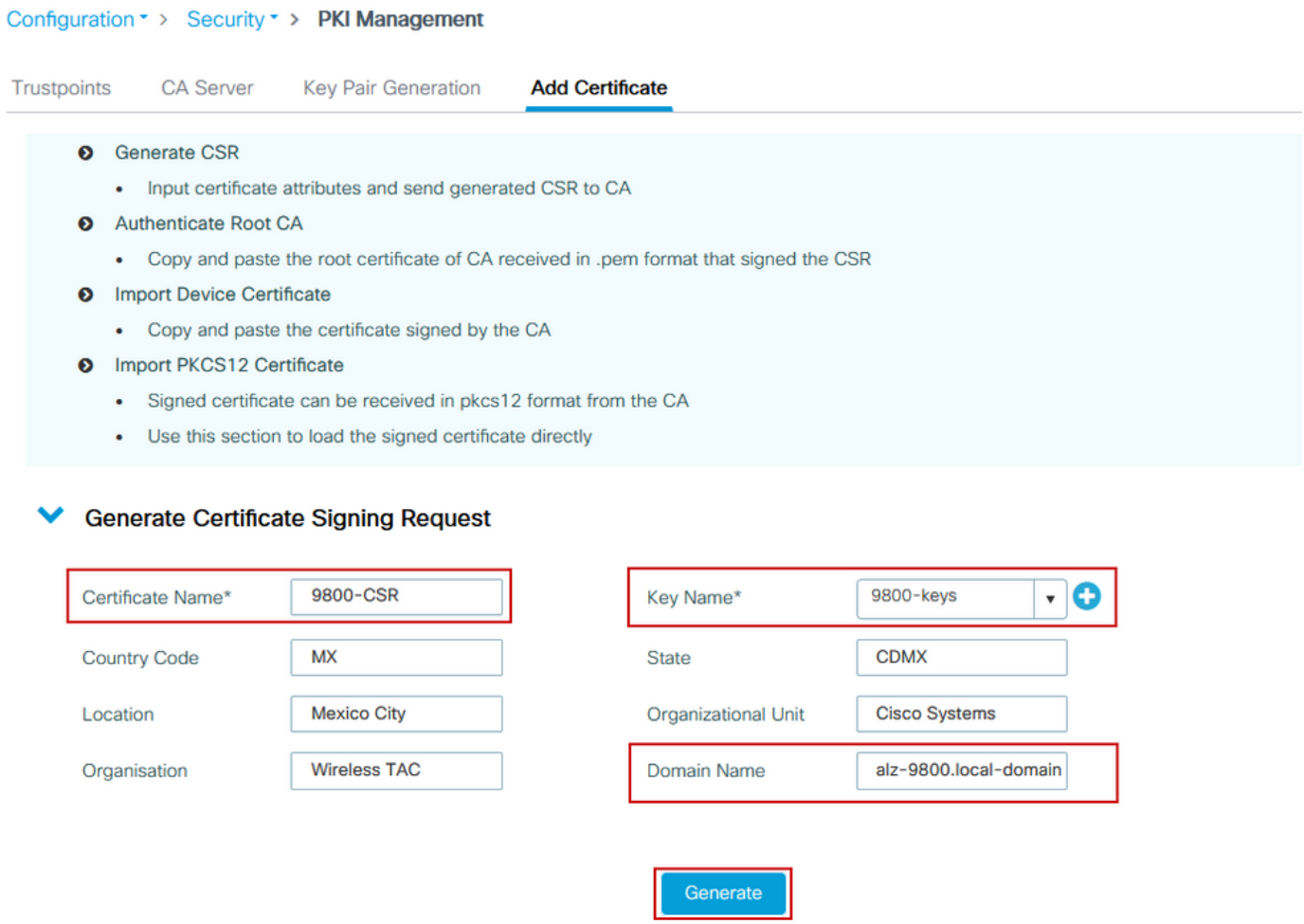
CLI 구성:

```
9800(config)#crypto key generate rsa general-keys label 9800-keys exportable
The name for the keys will be: 9800-keys
Choose the size of the key modulus in the range of 512 to 4096 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.

How many bits in the modulus [1024]: 4096
% Generating 4096 bit RSA keys, keys will be exportable...
[OK] (elapsed time was 9 seconds)
```

2단계.9800 WLC에 대한 CSR을 생성합니다. **Add Certificate(인증서 추가)** 탭으로 이동하고 **Generate Certificate Signing Request(인증서 서명 요청 생성)**를 확장하고 세부 정보를 입력한 다음 드롭다운 목록에서 이전에 생성한 키 쌍을 선택합니다. **도메인 이름**은 9800 WLC(웹 관리 페이지, 웹 인증 페이지 등)에서 클라이언트 액세스에 대해 정의된 URL과 일치해야 하며, **인증서 이름**은 신

뢰 지점 이름이므로 사용을 기준으로 이름을 지정할 수 있습니다.

> **참고**: 9800 WLC는 CN(Common Name) 내에 와일드카드 매개변수를 포함하는 인증서를 지원합니다.



정보가 올바른지 확인한 다음 Generate(생성)를 **클릭합니다**. 이렇게 하면 CSR이 원래 양식 옆에 있는 텍스트 상자에 표시됩니다.

복사는 텍스트 편집기에 붙여 넣고 CSR을 저장할 수 있도록 복사본을 클립보드에 저장합니다. Save **to device**(**디바이스에 저장**)를 선택하면 9800 WLC에서 CSR의 사본을 생성하여 bootflash:/csr에 저장합니다. 예를 들어 다음 명령을 실행합니다.

```
9800#dir bootflash:/csr
Directory of bootflash:/csr/

1046531 -rw- 1844 Sep 28 2021 18:33:49 +00:00 9800-CSR1632856570.csr
```

```
26458804224 bytes total (21492699136 bytes free)
9800#more bootflash:/csr/9800-CSR1632856570.csr
-----BEGIN CERTIFICATE REQUEST-----
<Certificate Request>
-----END CERTIFICATE REQUEST-----
```

CLI 구성:

```
9800(config)#crypto pki trustpoint 9800-CSR
9800(ca-trustpoint)#enrollment terminal pem
9800(ca-trustpoint)#revocation-check none
9800(ca-trustpoint)#subject-name C=MX, ST=CDMX, L=Mexico City, O=Cisco Systems, OU=Wireless TaC,
CN=alz-9800.local-domain.com
9800(ca-trustpoint)#rsakeypair 9800-keys
9800(ca-trustpoint)#subject-alt-name domain1.mydomain.com,domain2.mydomain.com
9800(ca-trustpoint)#exit

(config)#crypto pki enroll 9800-CSR
% Start certificate enrollment ..

% The subject name in the certificate will include: C=MX, ST=CDMX, L=Mexico City, O=Cisco
Systems, OU=Wireless TaC, CN=alz-9800.local-domain.com
% The subject name in the certificate will include: alz-9800
% Include the router serial number in the subject name? [yes/no]: no
% Include an IP address in the subject name? [no]: no
Display Certificate Request to terminal? [yes/no]: yes
Certificate Request follows:

-----BEGIN CERTIFICATE REQUEST-----
<Certificate Request>
-----END CERTIFICATE REQUEST-----

---End - This line not part of the certificate request---

Redisplay enrollment request? [yes/no]: no
```

주체 이름 구성에 사용할 수 있는 매개 변수:

C: 나라에서는 두 개의 대문자만 사용해야 합니다.

ST: 일부 주는 시/도 이름을 나타냅니다.

L: 위치 이름, 도시를 나타냅니다.

O: 회사명을 가리킵니다.

OU: Organizational Unit Name은 섹션을 참조할 수 있습니다.

CN: (Common Name)인증서가 발급되는 주체를 가리킵니다. 액세스할 특정 IP 주소(무선 관리 IP, 가상 IP 등) 또는 FQDN을 사용하여 구성된 호스트 이름을 지정해야 합니다.

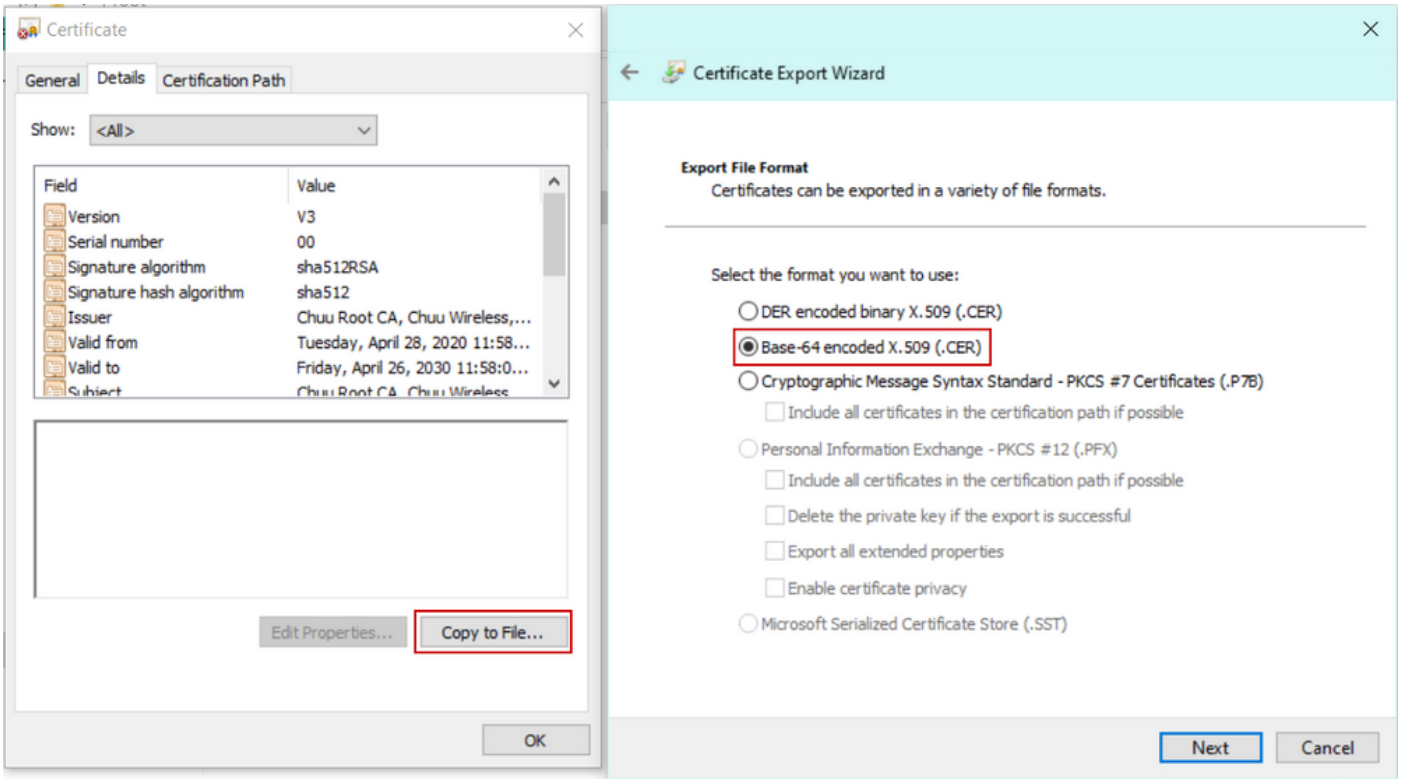
> **참고**: Subject Alternate Name(주체 대체 이름)을 추가하려면 Cisco 버그 ID CSCvt로 인해 17.8.1 이전 버전의 Cisco IOS XE에서 이 이름을 추가할 수 [없습니다15177](#) . 이 시나리오에서는 SAN이 없기 때문에 일부 브라우저 알림이 표시될 수 있습니다. 이를 방지하기 위해 옵션 1과 같이 키와 CSR을 오프박스로 생성합니다.

3단계. CA(Certification Authority)에서 서명한 CSR을 가져옵니다. 서명을 받으려면 전체 문자열을

CA로 전송해야 합니다.

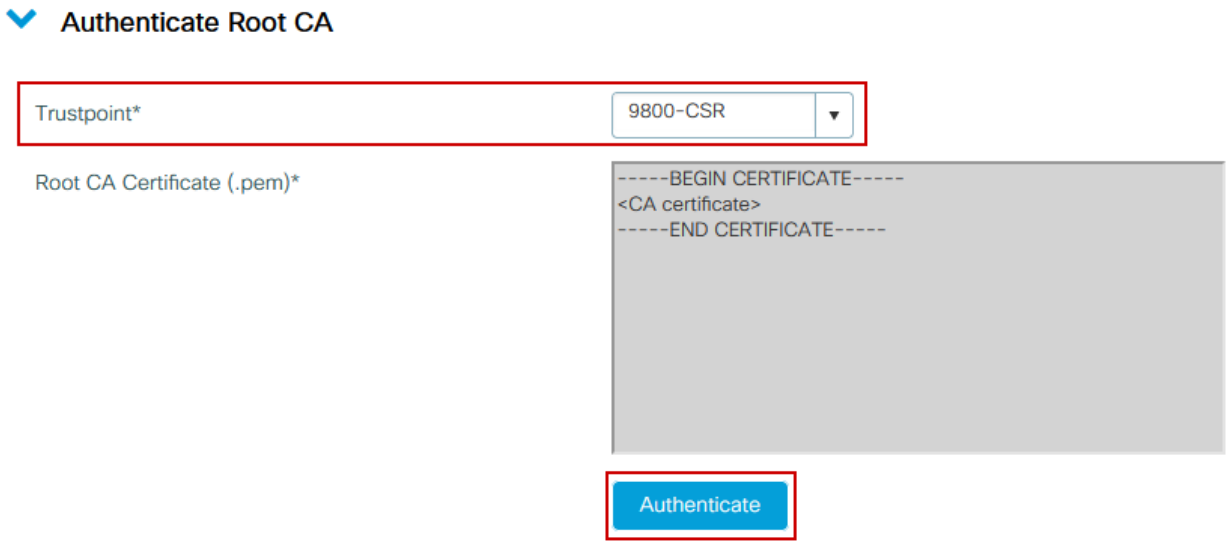
```
-----BEGIN CERTIFICATE REQUEST-----
<Certificate Request>
-----END CERTIFICATE REQUEST-----
```

Windows Server CA를 사용하여 인증서를 서명하는 경우 서명된 인증서를 Base64 형식으로 다운로드합니다. 그렇지 않으면 Windows cert manager와 같은 유틸리티를 사용하여 내보내야 합니다.



> **참고**: 신뢰 지점 인증 프로세스는 CSR에 서명한 CA 수에 따라 달라집니다. 단일 레벨 CA가 있는 경우 **4a단계를 선택합니다.** 다중 레벨 CA가 있는 경우 **4b단계로 이동합니다**. 이는 신뢰 지점에서 한 번에 두 개의 인증서(주체 인증서 및 발급자 인증서)만 저장할 수 있기 때문에 필요합니다.

4a단계. 9800에서 발급자 CA를 신뢰하도록 합니다. 발급자 CA 인증서를 .pem 형식(Base64)으로 다운로드합니다. 동일한 메뉴에서 **Authentication Root CA**(인증 루트 CA) 섹션을 확장하고 Trustpoint(신뢰 지점) 드롭다운 목록에서 이전에 정의된 **신뢰 지점**을 선택한 다음 발급자 CA 인증서를 붙여넣습니다. 세부 정보가 올바르게 구성되었는지 확인하고 Authenticate(인증)를 **클릭합니다**.



CLI 구성:

```
9800(config)# crypto pki authenticate 9800-CSR

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
```

```
Certificate has the following attributes: Fingerprint MD5: DD05391A 05B62573 A38C18DD CDA2337C
Fingerprint SHA1: 596DD2DC 4BF26768 CFB14546 BC992C3F F1408809 % Do you accept this certificate?
[yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
```

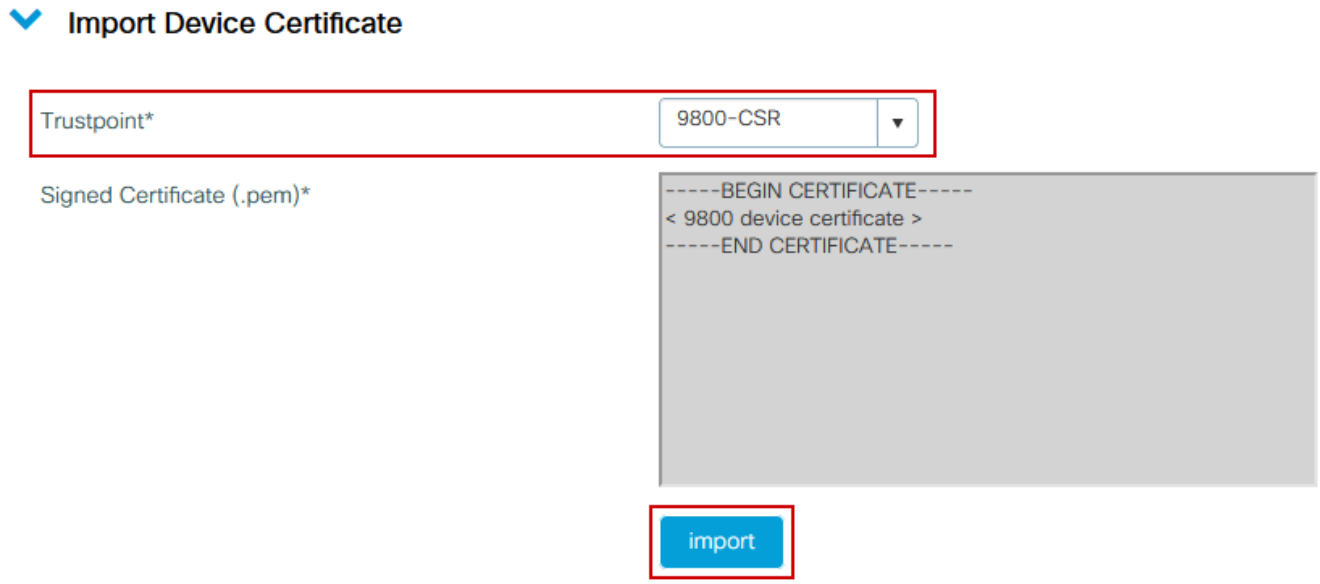
4b단계. 여러 권한 부여 레벨이 존재하는 시나리오에서 모든 CA 레벨에 새 신뢰 지점이 필요합니다. 이러한 신뢰 지점은 인증 인증서만 포함하고 다음 인증 수준을 가리킵니다. 이 프로세스는 CLI에서만 수행되며 이 예에서는 중간 CA 1개와 루트 CA 1개가 있습니다.

```
9800(config)#crypto pki trustpoint root
9800(ca-trustpoint)#enrollment terminal
9800(ca-trustpoint)#chain-validation stop
9800(ca-trustpoint)#revocation-check none
9800(ca-trustpoint)#exit
9800(config)#crypto pki authenticate root

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
```

```
-----END CERTIFICATE-----

Certificate has the following attributes:
Fingerprint MD5: 6CAC00D5 C5932D01 B514E413 D41B37A8
Fingerprint SHA1: 5ABD5667 26B7BD0D 83BDFC34 543297B7 3D3B3F24

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported

9800(config)#crypto pki trustpoint 9800-CSR
9800(ca-trustpoint)#chain-validation continue root
9800(config)#crypto pki authenticate 9800-CSR

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----




-----END CERTIFICATE-----

Certificate has the following attributes:
Fingerprint MD5: DD05391A 05B62573 A38C18DD CDA2337C
Fingerprint SHA1: 596DD2DC 4BF26768 CFB14546 BC992C3F F1408809
Certificate validated - Signed by existing trustpoint CA certificate.

Trustpoint CA certificate accepted.
% Certificate successfully imported
```

> **참고**: 인증 체인에 둘 이상의 중간 CA가 있는 경우 추가 인증 레벨별로 새 신뢰 지점을 생성해야 합니다. 이 신뢰 지점은 **chain-validation continue <trustpoint-name>** 명령을 사용하여 다음 인증 수준을 포함하는 신뢰 **지점을 참조해야** 합니다.

5단계. 서명된 인증서를 9800 WLC에 로드합니다. 동일한 메뉴에서 **Import Device Certificate(디바이스 인증서** 가져오기) 섹션을 확장합니다. 이전에 정의된 신뢰 **지점**을 선택하고 CA에서 제공한 서명된 디바이스 인증서를 붙여넣습니다. 그런 다음 인증서 **정보**가 확인되면 import(가져오기)를 클릭합니다.



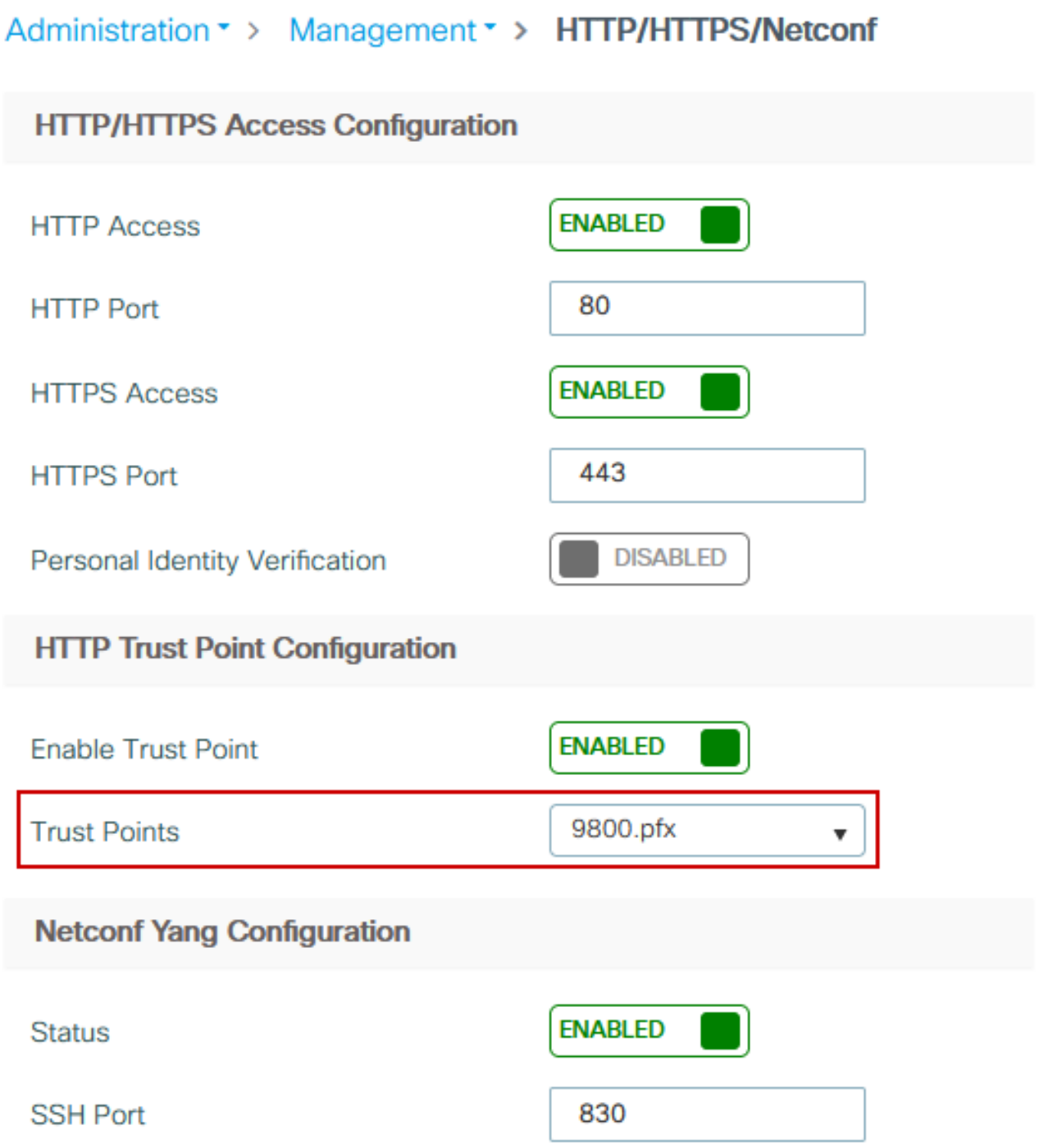
CLI 구성:

```
9800(config)#crypto pki import 9800-CSR certificate

Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
<9800 device certificate >
-----END CERTIFICATE-----

% Router Certificate successfully imported
```

새 인증서 사용

웹 관리

Administration > Management > HTTP/HTTPS/Netconf로 이동하고 **Trust Points** 드롭다운 목록에서 가져온 **인증서**를 선택합니다.

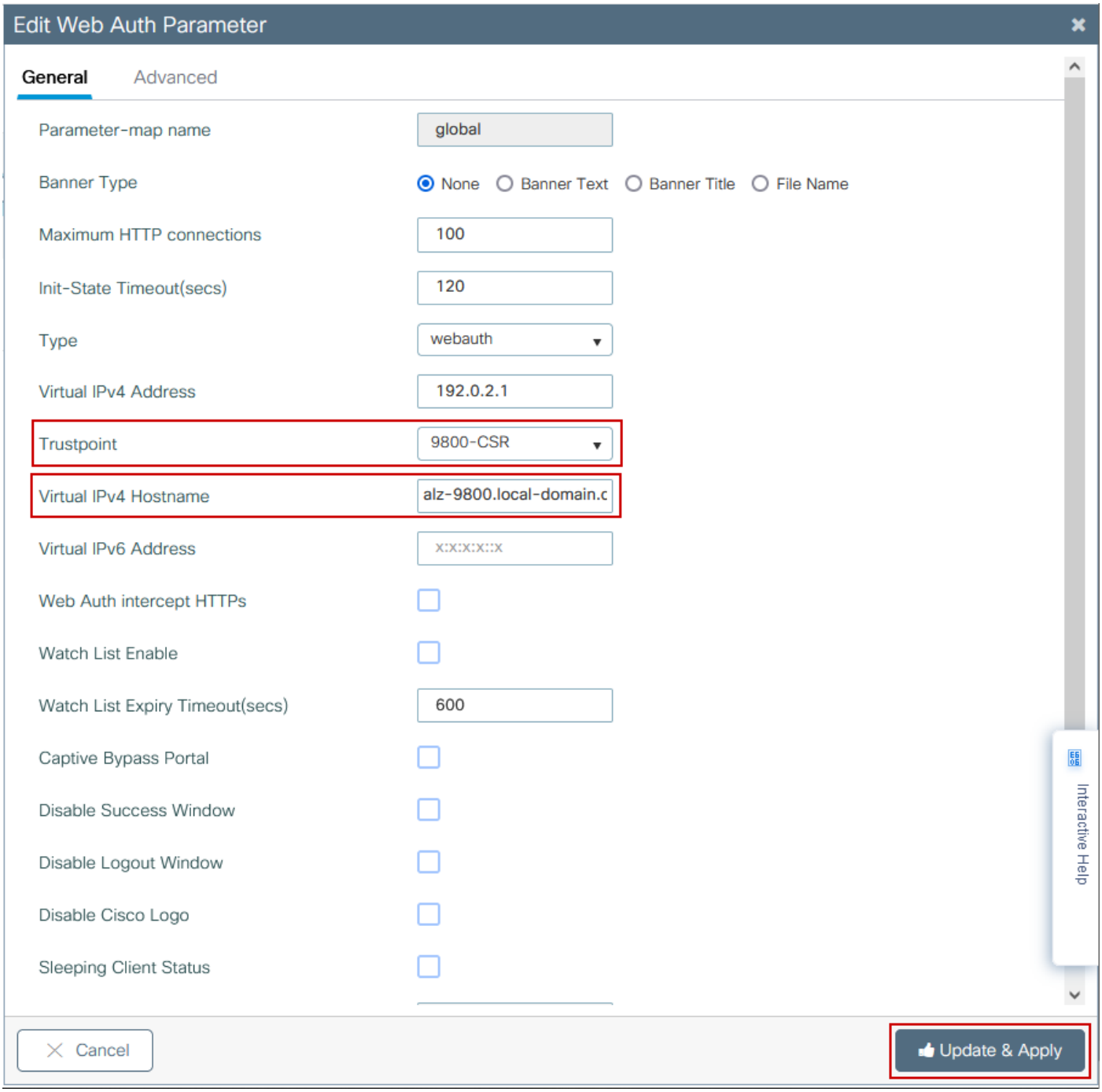


CLI 구성:

```
9800(config)#ip http secure-trustpoint 9800.pfx
9800(config)#no ip http secure-server
9800(config)#ip http secure-server
```

로컬 웹 인증

Configuration(**컨피그레이션**) > Security(**보안**) > Web Auth(**웹 인증**)로 이동하고 **전역** 매개변수 맵을 선택한 다음 Trustpoint(신뢰 지점) 드롭다운 목록에서 가져온 **신뢰 지점**을 선택합니다. Update & Apply(**업데이트 및** 적용)를 클릭하여 변경 사항을 저장합니다. **가상** IPv4 **호스트 이름**이 인증서의 일반 이름과 일치하는지 확인합니다.



CLI 구성:

```
9800(config)#parameter-map type webauth global
9800(config-params-parameter-map)#type webauth
9800(config-params-parameter-map)#virtual-ip ipv4 192.0.2.1 virtual-host alz-9800.local-domain.com
9800(config-params-parameter-map)#trustpoint 9800-CSR
```

인증서 사용을 업데이트하려면 HTTP 서비스를 다시 시작하십시오.

```
9800(config)#no ip http server
9800(config)#ip http server
```

.

고가용성 고려 사항

HA SSO(Stateful Switchover High Availability)를 위해 구성된 9800 쌍에서 모든 인증서는 초기 대량 동기화 시 기본 인증서에서 보조 인증서로 복제됩니다. 여기에는 RSA 키를 내보낼 수 없도록 구성한 경우에도 컨트롤러 자체에 개인 키가 생성된 인증서가 포함됩니다. HA 쌍이 설정되면 설치된 모든 새 인증서가 두 컨트롤러에 설치되고 모든 인증서가 실시간으로 복제됩니다.

장애가 발생하면 이전의 보조-현재-활성 컨트롤러는 기본 컨트롤러에서 상속된 인증서를 투명하게 사용합니다.

웹 브라우저에서 인증서를 신뢰할 수 있는지 확인하는 방법

웹 브라우저에서 인증서를 신뢰할 수 있도록 하려면 몇 가지 중요한 사항을 고려해야 합니다.

- CN(또는 SAN 필드)은 브라우저에서 방문한 URL과 일치해야 합니다.
- 유효기간 내에 있어야 합니다.
- CA 또는 CA 체인에 의해 발급되어야 하며, 이 체인의 루트는 브라우저에 의해 신뢰됩니다. 이를 위해 웹 서버에서 제공하는 인증서는 클라이언트 브라우저(일반적으로 루트 CA)에서 신뢰하는 인증서가 포함될 때까지(반드시 포함되지는 않음) 체인의 모든 인증서를 포함해야 합니다 .
- 폐기 목록이 포함된 경우 브라우저에서 폐기 목록을 다운로드할 수 있어야 하며 인증서 CN은 나열되지 않아야 합니다.

다음을 확인합니다.

다음 명령을 사용하여 인증서 컨피그레이션을 확인할 수 있습니다.

```
9800#show crypto pki certificate 9800.pfx

Certificate
Status: Available
Certificate Serial Number (hex): 1236
Certificate Usage: General Purpose
Issuer:
cn=Chuu Intermediate CA
ou=Chuu Wireless
o=Chuu Inc
st=CDMX
c=MX
Subject:
Name: alz-9800
e=user@example.com
cn=alz-9800
ou=Cisco Systems
o=Wireless TAC
l=CDMX
st=CDMX
c=MX
Validity Date:
start date: 17:54:45 Pacific Sep 28 2021
end date: 17:54:45 Pacific Sep 26 2031
Associated Trustpoints: 9800.pfx

CA Certificate
Status: Available
Certificate Serial Number (hex): 1000
Certificate Usage: Signature
```

```
Issuer:
cn=Chuu Root CA
ou=Chuu Wireless
o=Chuu Inc
l=Iztapalapa
st=CDMX
c=MX
Subject:
cn=Chuu Intermediate CA
ou=Chuu Wireless
o=Chuu Inc
st=CDMX
c=MX
Validity Date:
start date: 05:10:34 Pacific Apr 29 2020
end date: 05:10:34 Pacific Apr 27 2030
Associated Trustpoints: 9800.pfx
```

```
9800#show ip http server secure status

HTTP secure server status: Enabled
HTTP secure server port: 443
HTTP secure server ciphersuite: 3des-ede-cbc-sha aes-128-cbc-sha
aes-256-cbc-sha dhe-aes-128-cbc-sha ecdhe-rsa-3des-ede-cbc-sha
rsa-aes-cbc-sha2 rsa-aes-gcm-sha2 dhe-aes-cbc-sha2 dhe-aes-gcm-sha2
ecdhe-rsa-aes-cbc-sha2 ecdhe-rsa-aes-gcm-sha2
HTTP secure server TLS version: TLSv1.2 TLSv1.1 TLSv1.0
HTTP secure server client authentication: Disabled
HTTP secure server trustpoint: 9800.pfx
HTTP secure server active session modules: ALL
```

9800에서 인증서 체인을 확인할 수 있습니다. 중간 CA에서 발급한 디바이스 인증서, 즉 루트 CA에서 발급한 디바이스 인증서의 경우, 두 인증서로 구성된 그룹별로 하나의 신뢰 지점이 있으므로 각 레벨마다 고유한 신뢰 지점이 있습니다. 이 경우 9800 WLC에는 디바이스 인증서(WLC 인증서) 및 발급 CA(중간 CA)가 포함된 9800.pfx가 있습니다. 그런 다음 중간 CA를 발급한 루트 CA의 또 다른 신뢰 지점입니다.

```
9800#show crypto pki certificate 9800.pfx
Certificate
Status: Available
Certificate Serial Number (hex): 1236
Certificate Usage: General Purpose
Issuer:
cn=Chuu Intermediate CA
ou=Chuu Wireless
o=Chuu Inc
st=CDMX
c=MX
Subject:
Name: alz-9800
e=user@example.com
cn=alz-9800
ou=Cisco Systems
o=Wireless TAC
l=CDMX
st=CDMX
c=MX
Validity Date:
start date: 17:54:45 Pacific Sep 28 2021
end date: 17:54:45 Pacific Sep 26 2031
Associated Trustpoints: 9800.pfx
```

```
CA Certificate
Status: Available
Certificate Serial Number (hex): 1000
Certificate Usage: Signature
Issuer:
cn=Chuu Root CA
ou=Chuu Wireless
o=Chuu Inc
l=Iztapalapa
st=CDMX
c=MX
Subject:
cn=Chuu Intermediate CA
ou=Chuu Wireless
o=Chuu Inc
st=CDMX
c=MX
Validity Date:
start date: 05:10:34 Pacific Apr 29 2020
end date: 05:10:34 Pacific Apr 27 2030
Associated Trustpoints: 9800.pfx

9800#show crypto pki certificate 9800.pfx-rrr1
CA Certificate
Status: Available
Certificate Serial Number (hex): 00
Certificate Usage: Signature
Issuer:
cn=Chuu Root CA
ou=Chuu Wireless
o=Chuu Inc
l=Iztapalapa
st=CDMX
c=MX
Subject:
cn=Chuu Root CA
ou=Chuu Wireless
o=Chuu Inc
l=Iztapalapa
st=CDMX
c=MX
Validity Date:
start date: 04:58:05 Pacific Apr 29 2020
end date: 04:58:05 Pacific Apr 27 2030
Associated Trustpoints: 9800-CSR 9800.pfx-rrr1
```

OpenSSL을 사용한 인증서 확인

OpenSSL은 인증서 자체를 확인하거나 일부 변환 작업을 수행하는 데 유용할 수 있습니다.

OpenSSL을 사용하는 인증서를 표시하려면

```
openssl x509 -in
```

CSR의 콘텐츠를 표시하려면

openssl req -noout -text -in

9800 WLC에서 최종 인증서를 확인하고 싶지만 브라우저 이외의 다른 것을 사용하려는 경우 OpenSSL에서 이를 수행할 수 있으며 자세한 정보를 제공합니다.

openssl s_client -showcerts -verify 5 -connect

<wlcURL>을 9800의 webadmin URL 또는 게스트 포털(가상 IP)의 URL로 교체할 수 있습니다. 또한 IP 주소를 입력할 수 있습니다. 어떤 인증서 체인이 수신되는지 알려주지만 호스트 이름 대신 IP 주소를 사용할 경우 인증서 검증은 100% 정확할 수 없습니다.

내용을 보고 PKCS12(.pfx) 인증서 또는 인증서 체인을 확인하려면 다음을 수행합니다.

openssl pkcs12 -info -in

다음은 장치 인증서가 "intermediate.com"이라는 중간 CA에 의해 TAC(Technical Assistance Center)로 발급되고 그 자체가 "root.com"이라는 루트 CA에 의해 발급되는 인증서 체인에 대한 이 명령의 예입니다.

```
openssl pkcs12 -info -in chainscript2.pfx
Enter Import Password:
MAC Iteration 2048
MAC verified OK
PKCS7 Encrypted data: pbeWithSHA1And40BitRC2-CBC, Iteration 2048
Certificate bag
Bag Attributes
localKeyID: 1D 36 8F C2 4B 18 0B 0D B2 57 A2 55 18 96 7A 8B 57 F9 CD FD
subject=/C=BE/ST=Diegem/L=Diegem/O=Cisco/CN=TAC
issuer=/C=BE/ST=Diegem/O=Cisco/OU=TAC/CN=intermediate.com/emailAddress=int@int.com
-----BEGIN CERTIFICATE-----
<Device certificate >
-----END CERTIFICATE-----
Certificate bag
Bag Attributes: <No Attributes>
subject=/C=BE/ST=Diegem/O=Cisco/OU=TAC/CN=intermediate.com/emailAddress=int@int.com
issuer=/C=BE/ST=Diegem/L=Diegem/O=Cisco/OU=TAC/CN=RootCA.root.com/emailAddress=root@root.com
-----BEGIN CERTIFICATE-----
<Intermediate certificate >
-----END CERTIFICATE-----
Certificate bag
Bag Attributes: <No Attributes>
subject=/C=BE/ST=Diegem/L=Diegem/O=Cisco/OU=TAC/CN=RootCA.root.com/emailAddress=root@root.com
issuer=/C=BE/ST=Diegem/L=Diegem/O=Cisco/OU=TAC/CN=RootCA.root.com/emailAddress=root@root.com
-----BEGIN CERTIFICATE-----
<Root certificate >
```

```
-----END CERTIFICATE-----
PKCS7 Data
Shrouded Keybag: pbeWithSHA1And3-KeyTripleDES-CBC, Iteration 2048
Bag Attributes
localKeyID: 1D 36 8F C2 4B 18 0B 0D B2 57 A2 55 18 96 7A 8B 57 F9 CD FD
Key Attributes: <No Attributes>
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----BEGIN ENCRYPTED PRIVATE KEY-----
<Private key >
-----END ENCRYPTED PRIVATE KEY-----
```

문제 해결

문제를 해결하려면 이 명령을 사용합니다. 원격 세션(SSH 또는 텔넷)에서 수행하는 경우 **터미널 모니터**가 출력을 표시해야 합니다.

```
9800#debug crypto pki transactions
```

성공적인 시나리오 디버그 출력

이 출력은 9800에서 인증서 가져오기가 성공할 때의 예상 출력을 표시합니다. 이 정보를 참조하여 오류 상태를 확인합니다.

```
Sep 28 17:35:23.242: CRYPTO_PKI: Copying pkcs12 from bootflash:9800.pfx
Sep 28 17:35:23.322: CRYPTO_PKI: Creating trustpoint 9800.pfx
Sep 28 17:35:23.322: %PKI-6-TRUSTPOINT_CREATE: Trustpoint: 9800.pfx created succesfully
Sep 28 17:35:23.324: CRYPTO_PKI: examining cert:
Sep 28 17:35:23.324: CRYPTO_PKI: issuerName=cn=Chuu Intermediate CA,ou=Chuu Wireless,o=Chuu
Inc,st=CDMX,c=MX
Sep 28 17:35:23.324: CRYPTO_PKI: subjectname=e=user@example.com,cn=alz-9800,ou=Cisco
Systems,o=Wireless TAC,l=CDMX,st=CDMX,c=MX
Sep 28 17:35:23.324: CRYPTO_PKI: adding RSA Keypair
Sep 28 17:35:23.324: CRYPTO_PKI: bitValue of ET_KEY_USAGE = 140
Sep 28 17:35:23.324: CRYPTO_PKI: Certificate Key Usage = GENERAL_PURPOSE
Sep 28 17:35:23.324: %CRYPTO_ENGINE-5-KEY_ADDITION: A key named 9800.pfx has been generated or
imported by pki-pkcs12
Sep 28 17:35:23.331: CRYPTO_PKI: adding as a router certificate.Public key in cert and stored
public key 9800.pfx match

Sep 28 17:35:23.333: CRYPTO_PKI: examining cert:
Sep 28 17:35:23.333: CRYPTO_PKI: issuerName=cn=Chuu Root CA,ou=Chuu Wireless,o=Chuu
Inc,l=Iztapalapa,st=CDMX,c=MX
Sep 28 17:35:23.333: CRYPTO_PKI: subjectname=cn=Chuu Intermediate CA,ou=Chuu Wireless,o=Chuu
Inc,st=CDMX,c=MX
Sep 28 17:35:23.333: CRYPTO_PKI: no matching private key presents.

[...]

Sep 28 17:35:23.335: CRYPTO_PKI: Setting the key_type as RSA
Sep 28 17:35:23.335: CRYPTO_PKI: Attempting to insert the peer's public key into cache
Sep 28 17:35:23.335: CRYPTO_PKI:Peer's public inserted successfully with key id 21
Sep 28 17:35:23.336: Calling pkiSendCertInstallTrap to send alert
Sep 28 17:35:23.337: CRYPTO_PKI: Deleting cached key having key id 31
Sep 28 17:35:23.337: CRYPTO_PKI: Attempting to insert the peer's public key into cache
Sep 28 17:35:23.337: CRYPTO_PKI:Peer's public inserted successfully with key id 32
Sep 28 17:35:23.338: CRYPTO_PKI: (A0323) Session started - identity selected (9800.pfx)
Sep 28 17:35:23.338: CRYPTO_PKI: Rcvd request to end PKI session A0323.
```

```
Sep 28 17:35:23.338: CRYPTO_PKI
alz-9800#: PKI session A0323 has ended. Freeing all resources.
Sep 28 17:35:23.338: CRYPTO_PKI: unlocked trustpoint 9800.pfx, refcount is 0
Sep 28 17:35:23.338: CRYPTO_PKI: Expiring peer's cached key with key id 32Public key in cert and
stored public key 9800.pfx match

Sep 28 17:35:23.341: Calling pkiSendCertInstallTrap to send alert
Sep 28 17:35:23.341: CRYPTO_PKI: cert verified and inserted.
Sep 28 17:35:23.402: CRYPTO_PKI: Creating trustpoint 9800.pfx-rrr1
Sep 28 17:35:23.402: %PKI-6-TRUSTPOINT_CREATE: Trustpoint: 9800.pfx-rrr1 created succesfully
Sep 28 17:35:23.403: CRYPTO_PKI: Setting the key_type as RSA
Sep 28 17:35:23.404: CRYPTO_PKI: Attempting to insert the peer's public key into cache
Sep 28 17:35:23.404: CRYPTO_PKI:Peer's public inserted successfully with key id 22
Sep 28 17:35:23.405: Calling pkiSendCertInstallTrap to send alert
Sep 28 17:35:23.406: CRYPTO_PKI: no CRLs present (expected)
Sep 28 17:35:23.406: %PKI-6-PKCS12_IMPORT_SUCCESS: PKCS #12 import in to trustpoint 9800.pfx
successfully imported.
```

CA가 없는 PKCS12 인증서 가져오기 시도

인증서를 가져오고 "CA cert is not found." 오류가 발생하는 경우 .pfx 파일에 전체 체인이 없거나 CA 하나가 없는 것입니다.

```
9800(config)#crypto pki import pkcs12.pfx pkcs12 bootflash:pks12.pfx password
```

```
% Importing pkcs12...
Source filename [pks12.pfx]?
Reading file from bootflash:pks12.pfx
% Warning: CA cert is not found. The imported certs might not be usable.
```

openssl pkcs12 -info -in <path to cert> 명령을 실행하고 개인 키가 하나인 인증서가 하나만 표시되는 경우 CA가 없는 것입니다. 일반적으로 이 명령은 전체 인증서 체인을 나열하는 것이 좋습니다. 최상위 루트 CA가 이미 클라이언트 브라우저에 알려진 경우 이를 포함할 필요가 없습니다.

이를 해결하기 위한 한 가지 방법은 PKCS12를 PEM으로 해체하고 체인을 적절하게 재구축하는 것입니다. 다음 예에서는 디바이스(WLC) 인증서 및 키만 포함된 .pfx 파일이 있었습니다. 중간 CA(PKCS12 파일에 없음)가 발급했으며, 이는 다시 잘 알려진 루트 CA에 의해 서명되었습니다.

1단계. 개인 키를 내보냅니다.

```
openssl pkcs12 -in
```

2단계. 인증서를 PEM으로 내보냅니다.

```
openssl pkcs12 -in
```

3단계. 중간 CA 인증서를 PEM으로 다운로드합니다.

CA의 소스는 해당 CA의 성격에 따라 달라집니다. 공용 CA인 경우 온라인 검색으로 저장소를 찾을 수 있습니다. 그렇지 않으면 CA 관리자가 Base64 형식(.pem)으로 인증서를 제공해야 합니다. 여러 레벨의 CA가 있는 경우 **옵션 1** 가져오기 프로세스의 끝에 표시된 것과 같은 단일 파일로 그룹화합니다.

4단계. 키, 디바이스 인증서 및 CA 인증서에서 PKCS 12를 재구축합니다.

```
openssl pkcs12 -export -out fixedcertchain.pfx -inkey cert.key -in certificate.pem -certfile
CA.pem
```

이제 Catalyst 9800으로 손쉽게 가져올 수 있는 "fixedcertchain.pfx"가 있습니다!

참고 및 제한 사항

- Cisco IOS® XE는 2099년 이후의 유효한 CA 인증서를 지원하지 않습니다. Cisco 버그 ID CSCvp64208
- Cisco IOS® XE는 SHA256 메시지 다이제스트 PKCS 12 번들을 지원하지 않습니다(SHA256 인증서가 지원되지만 PKCS12 번들 자체가 SHA256으로 서명된 경우에는 지원되지 않음). Cisco 버그 ID CSCvz41428
- WLC에서 사용자 인증서를 전달해야 하는 경우 그리고 NAC/ISE 어플라이언스가 인터넷을 통해 연결 가능한 경우(예: SD-WAN 구축)에는 프래그먼트화가 표시됩니다. 인증서는 거의 항상 1500바이트보다 큽니다(즉, 인증서 메시지를 전달하기 위해 여러 RADIUS 패킷이 전송됨). 네트워크 경로에 여러 MTU가 있는 경우 RADIUS 패킷 자체의 조각화가 발생할 수 있습니다. 이러한 경우 인터넷 날씨로 인해 발생할 수 있는 지연/지터와 같은 문제를 방지하려면 동일한 경로를 통해 WLC 트래픽에 대한 모든 UDP 데이터그램을 전송하는 것이 좋습니다

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.