CSR-certificaten genereren en downloaden op Catalyst 9800 WLC's

Inhoud

Inleiding

Dit document beschrijft het algemene proces voor het genereren, downloaden en installeren van certificaten op de Catalyst 9800

Voorwaarden

Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- Hoe de 9800 WLC, het access point (AP) te configureren voor basisbediening
- De OpenSSL-toepassing gebruiken
- Public Key Infrastructure (PKI) en digitale certificaten

Gebruikte componenten

De informatie in dit document is gebaseerd op de volgende software- en hardware-versies:

- Catalyst 9800-L, Cisco IOS® XE versie 17.3.3
- OpenSSL-toepassing

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u zorgen dat u de potentiële impact van elke opdracht begrijpt.

Configureren

Op 16.10.X, 9800s niet ondersteunen een ander certificaat voor webverificatie en webbeheer. Het webloginportal gebruikt altijd het standaardcertificaat.

Op 16.11.X, kunt u een specifiek certificaat voor webverificatie configureren, het trustpoint definiëren binnen de globale parameter-map.

Er zijn twee opties om een certificaat te krijgen voor een 9800 WLC.

1. Genereren Certificaat-Ondertekeningsaanvraag (CSR) met OpenSSL of een andere SSL-toepassing. Ontvang een PKCS12-certificaat dat is ondertekend door uw certificeringsinstantie (CA) en laad het rechtstreeks naar de 9800 WLC. Dit betekent dat de privésleutel is gebundeld met dat certificaat.
2. Gebruik de 9800 WLC CLI om een CSR, krijg het ondertekend door een CA en laad vervolgens elk certificaat in de keten handmatig naar de 9800 WLC.

Gebruik degene die het beste bij uw behoeften past.

Optie 1 - Een reeds bestaand PKCS12-ondertekend certificaat laden

Een ondertekeningsaanvraag definiëren

Als u het certificaat nog niet hebt, moet u een ondertekeningsverzoek genereren om aan uw CA te geven.

Bewerk **openssl.cnf-**bestand vanuit uw huidige map (op een laptop waarop OpenSSL is geïnstalleerd), kopieer en plak deze regels om het veld Onderwerp Alternatieve Namen (SAN) op te nemen in nieuwe CSR's.

```
[ 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
```

Vervang de DNS.X-namen door uw SAN. Vervang de belangrijkste velden met de gewenste certificaatdetails. Zorg ervoor dat u de algemene naam in de SAN-velden (DNS.x) herhaalt. Google Chrome vereist dat de naam in de URL in de SAN-velden staat om het certificaat te kunnen vertrouwen.

In het geval van web admin moet u ook SAN-velden invullen met variaties van de URL (bijvoorbeeld alleen hostnaam of volledige Fully Qualified Domain Name (FQDN)) zodat het certificaat aansluit op wat de admin typt in de URL in de adresbalk van de browser.

Genereert de CSR via OpenSSL met deze opdracht:

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

De CSR genereert als **myCSR.csr** en zijn sleutel als **private.key** in de directory waar OpenSSL wordt uitgevoerd, tenzij het volledige pad naar de opdracht is voorzien.

Zorg ervoor om het **private.key** bestand veilig te houden aangezien het wordt gebruikt om communicatie te versleutelen.

U kunt de inhoud verifiëren met:

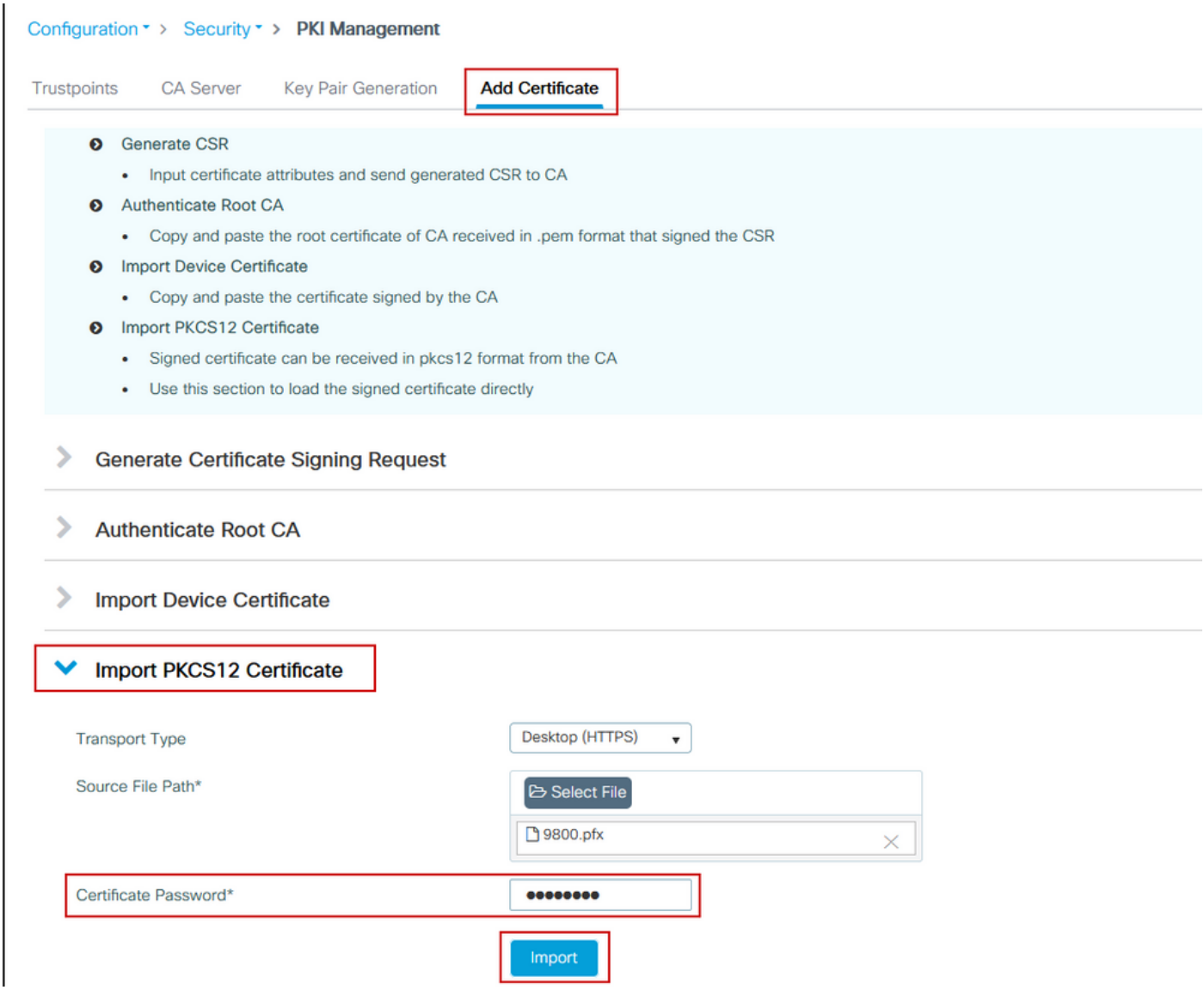
```
openssl req -noout -text -in myCSR.csr
```

U kunt dit MVO dan aan uw CA verstrekken om het ondertekend te hebben en een certificaat terug te ontvangen. Zorg ervoor dat de volledige keten van CA wordt gedownload en dat het certificaat in Base64-formaat is voor het geval het verdere manipulatie nodig heeft.

Certificaat importeren

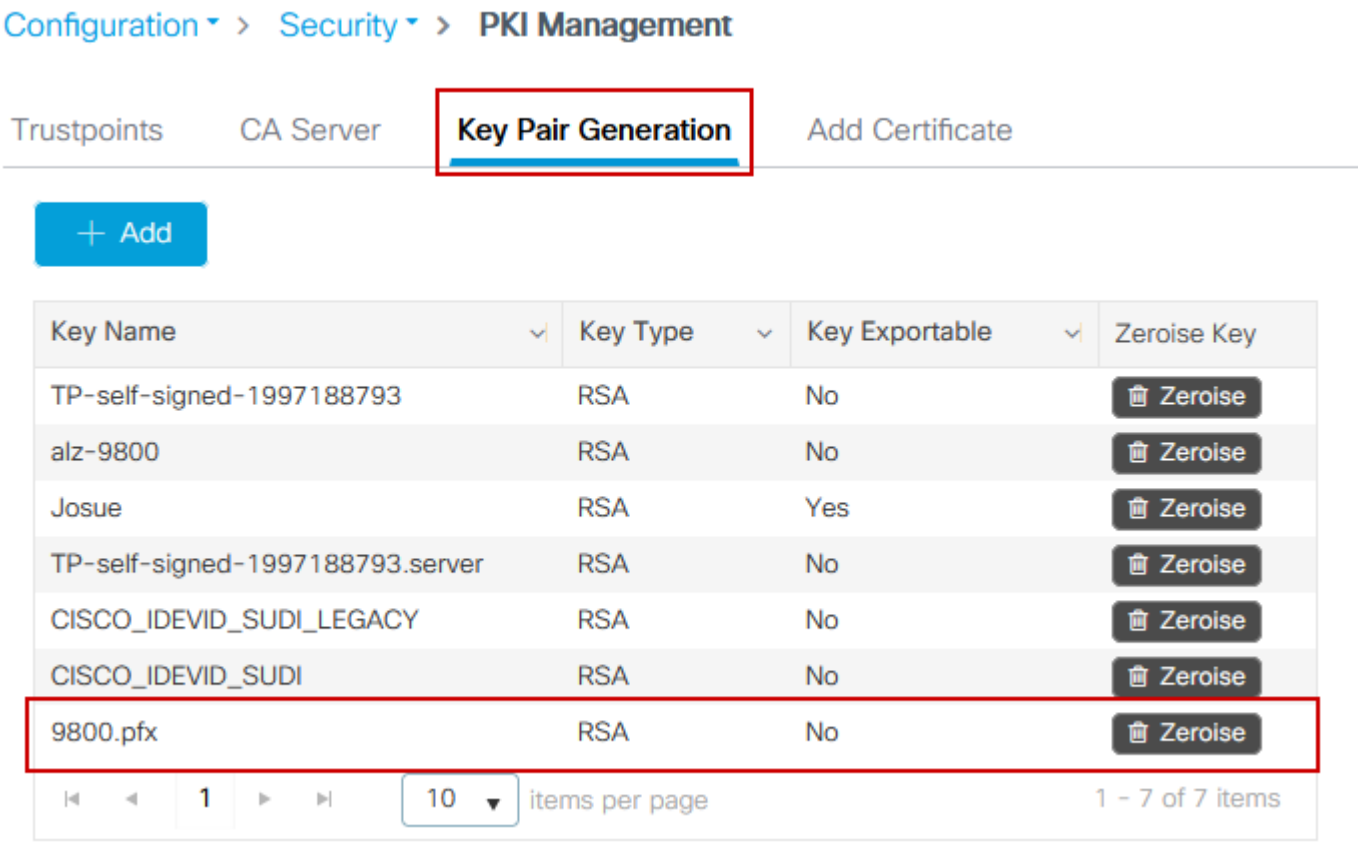
Stap 1. Sla uw PKCS12-certificaat op een Trivial File Transfer Protocol (TFTP)-server die bereikbaar is vanaf de 9800 WLC. Het PKCS12-certificaat moet de private sleutel en de certificaatketen tot de root CA bevatten.

Stap 2. Open uw 9800 WLC GUI en navigeer naar **Configuration > Security > PKI Management**, klik op het tabblad **Certificaat toevoegen.** Breid het menu **Certificaat PKCS12 importeren uit** en vul de TFTP-gegevens in. Alternatief, staat de optie van de **Desktop (HTTPS)** in de vervolgkeuzelijst van het **Type van Vervoer** HTTP toe uploadt door browser. **Het Wachtwoord voor** certificaten verwijst naar het wachtwoord dat werd gebruikt toen het PKCS12-certificaat werd gegenereerd.



Stap 3. Controleer of de informatie juist is en klik op **Importeren.** Daarna ziet u het nieuwe certificaat sleutelpaar voor dit nieuwe trustpoint geïnstalleerd in het tabblad **Key Pair Generation**. Bij een succesvolle import creëert de 9800 WLC ook een extra vertrouwenspunt voor meerlaagse CA's.

> **Opmerking**: De 9800 WLC presenteert momenteel niet de volledige certificaatketen wanneer een specifiek trustpoint wordt gebruikt voor webauth of webadmin, maar presenteert het apparaatcertificaat en de directe emittent ervan. Dit wordt gevolgd met Cisco fout-id CSCwa23606 , vast in Cisco IOS® XE 17.8.



CLI:

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

Opmerking: het is belangrijk dat zowel de certificaatbestandsnaam als de trustpoint naam exact overeenkomen voor de 9800 WLC om extra trustpoints te maken voor meerdere CA's.

PKCS12 Formaatconversie en certificaatketen in meerlaagse CA-scenario's.

Het is mogelijk om in een situatie te eindigen waar u een privaat sleutelbestand en certificaat in PEM of CRT formaat hebt en hen in een formaat willen combineren PKCS12 (.pfx) om aan 9800 WLC te uploaden. Voer deze opdracht in om dit te doen:

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

In het geval waar u een keten van certificaat (een of meerdere tussenliggende CA en root CA) allemaal in PEM-formaat hebt, moet u vervolgens alles combineren in één .pfx-bestand.

Combineer eerst handmatig de CA-certificaten in één bestand als zodanig. Kopieer en plak de inhoud samen (sla het bestand op in .pem-indeling):

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

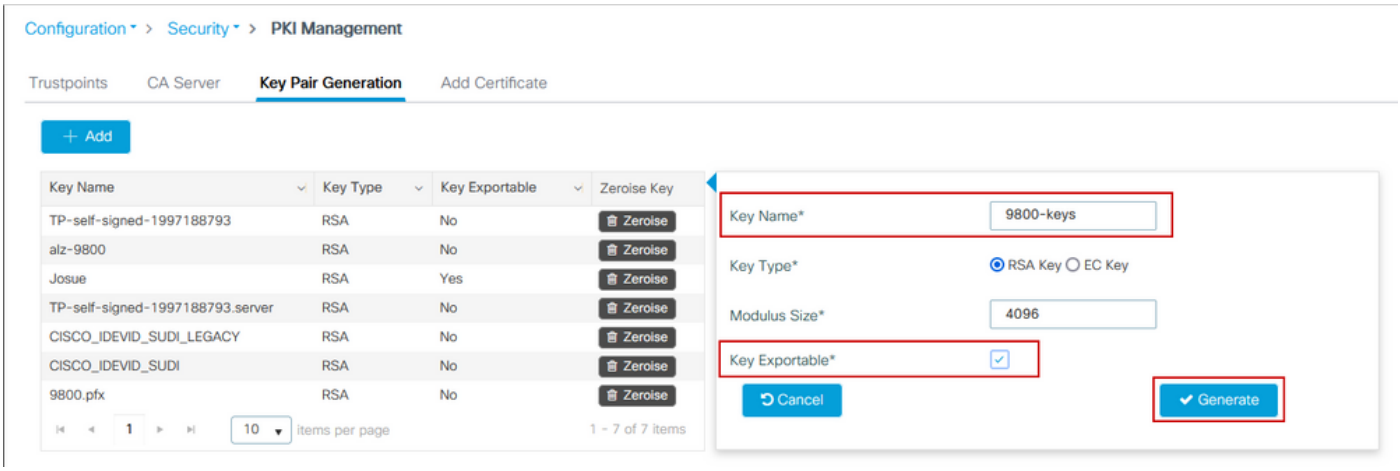
Later kunt u allen in één PKCS12 certificaatdossier met combineren:

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

Raadpleeg het gedeelte Verifiëren aan het einde van het artikel om te zien hoe het definitieve certificaat eruitziet.

Optie 2 - Een sleutel- en ondertekeningsaanvraag (CSR) definiëren op de 9800 WLC

Stap 1. Genereert een algemeen RSA-sleutelpaar. Navigeer naar **Configuratie > Beveiliging > PKI-beheer**, kies het tabblad **Key Pair Generation** en klik vervolgens op **+ Add**. Voer de gegevens in, zorg ervoor dat het aanvinkvakje **Key Exporteerbaar** is ingeschakeld en klik vervolgens op **Generate**.



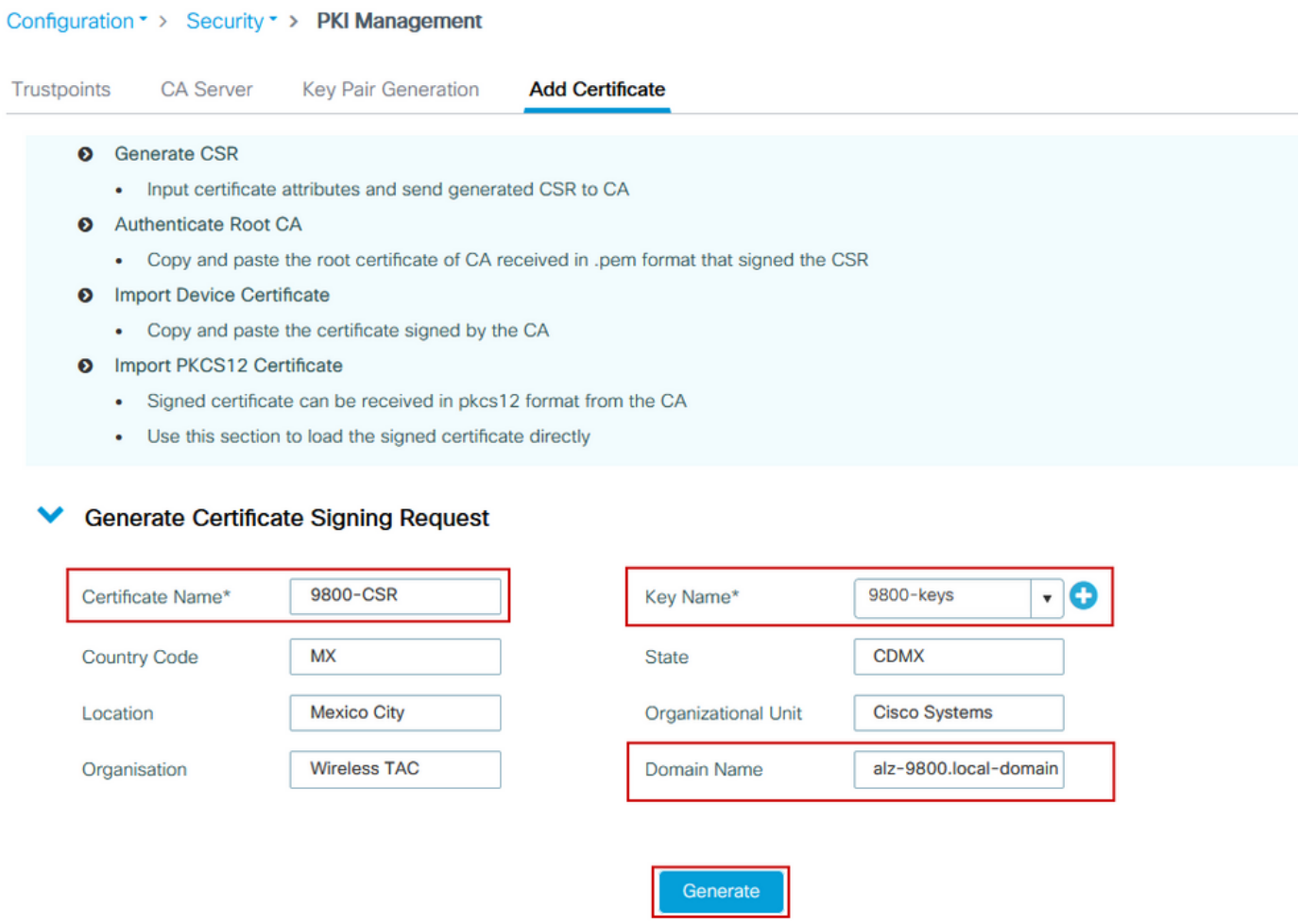
CLI-configuratie:

```
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)
```

Stap 2. Genereer een CSR voor uw 9800 WLC. Navigeer naar het tabblad **Certificaat toevoegen**

en vouw **Generate Certificate Signing Aanvraag uit**, vul de details in en kies het eerder gemaakte sleutelpaar uit de vervolgkeuzelijst. Het is belangrijk dat **Domeinnaam** overeenkomt met de URL die is gedefinieerd voor clienttoegang op de 9800 WLC (web admin pagina, web authenticatie pagina, enzovoort), **Certificaatnaam** is de trustpoint naam zodat u kunt benoemen op basis van het gebruik ervan.

> **Opmerking**: De 9800 WLC's ondersteunen certificaten met wildcard parameters binnen hun algemene naam.



Zorg ervoor dat de informatie correct is en klik vervolgens op **Generate**. Dit geeft de MVO weer in een tekstvak naast het originele formulier.

Kopiëren slaat een kopie naar het klembord op, zodat u deze in een teksteditor kunt plakken en de CSR kunt opslaan. Als **Save to device** is geselecteerd, maakt de 9800 WLC een kopie van de CSR en slaat deze op in **bootflash:**/csr. Voer bijvoorbeeld de volgende opdrachten uit:

```
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-configuratie:

```
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
```

Beschikbare parameters voor configuratie van onderwerpnaam:

C: Land, het moet slechts twee hoofdletters zijn.

ST: Sommige staat, verwijst naar staat of provincienaam.

L: Location Name, verwijst naar de stad.

O: Organisatienaam, verwijst naar bedrijf.

OU: Organisatorische Eenheid Naam, kan verwijzen naar sectie.

CN: (algemene naam)Verwijst naar het onderwerp waarop het certificaat is afgegeven, u moet ofwel het specifieke IP-adres opgeven dat moet worden geopend (draadloos IP-beheer, virtueel IP, enzovoort) of de geconfigureerde hostnaam met FQDN.

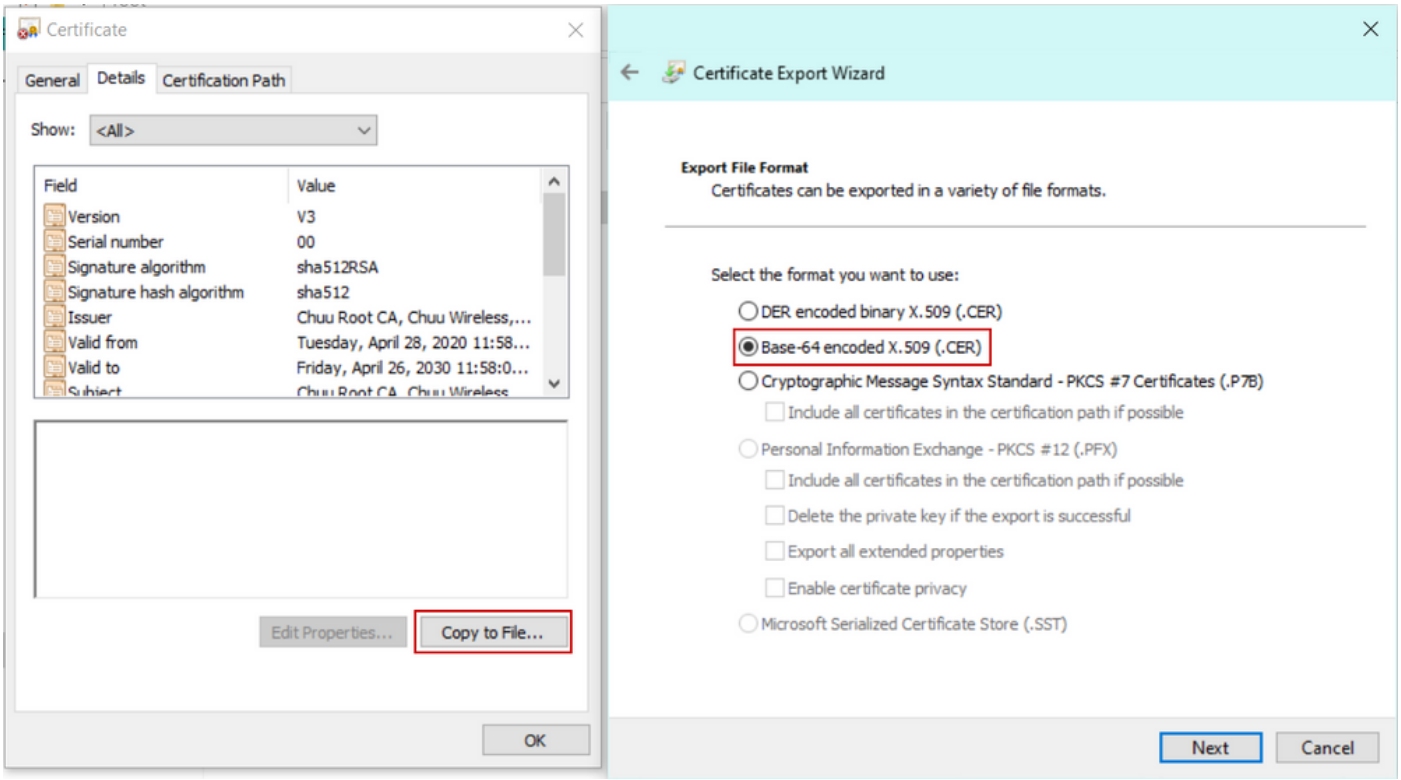
> **Opmerking**: als u een alternatieve onderwerpnaam wilt toevoegen, is dit niet mogelijk op

Cisco IOS XE-versies vóór 17.8.1 vanwege Cisco bug-id CSCvt15177 . Dit scenario kan resulteren in sommige browser waarschuwingen als gevolg van SAN niet aanwezig, om te voorkomen dat dit dan maken van de sleutel en CSR off-box zoals getoond in Optie 1.

Stap 3. Laat uw CSR ondertekenen door uw certificeringsinstantie (CA). De volledige string moet naar de CA gestuurd worden om het getekend te krijgen.

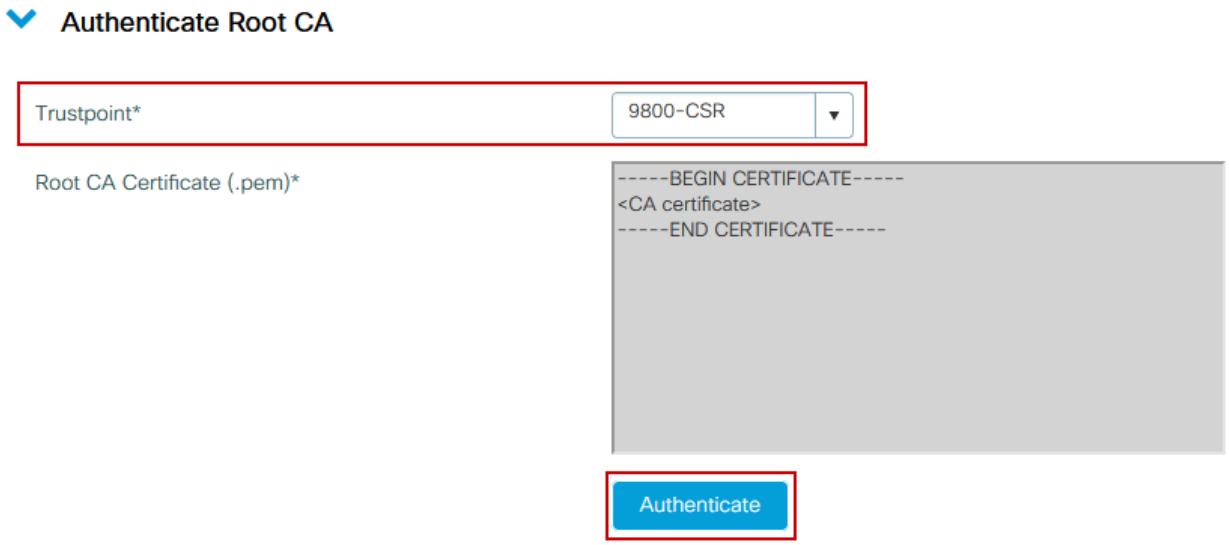
```
-----BEGIN CERTIFICATE REQUEST-----
<Certificate Request>
-----END CERTIFICATE REQUEST-----
```

Als u een Windows Server CA gebruikt om het certificaat te ondertekenen, downloadt u het ondertekende certificaat in Base64-indeling. Anders moet u exporteren met hulpprogramma's zoals Windows cert Manager.



Opmerking: het verificatieproces van trustpoint is afhankelijk van het aantal CA's dat uw CSR heeft ondertekend. Als er CA op één niveau is, controleer dan **Stap 4a.** Als er meerdere niveaus CA zijn, ga dan naar **Stap 4b**. Dit is nodig omdat een trustpoint slechts twee certificaten tegelijkertijd kan opslaan (het onderwerpcertificaat en het emittentencertificaat).

Stap 4a. Maak 9800 vertrouwen de emittent CA. Download het CA certificaat van de uitgever in .pem formaat (Base64). Breid de **CA-**sectie van de **verificatieroot** binnen hetzelfde menu uit, kies het eerder gedefinieerde trustpoint uit de vervolgkeuzelijst van **Trustpoint** en plak het CA-certificaat van de emittent. Zorg ervoor dat de gegevens correct zijn geconfigureerd en klik op **Verifiëren**.



CLI-configuratie:

```
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
```

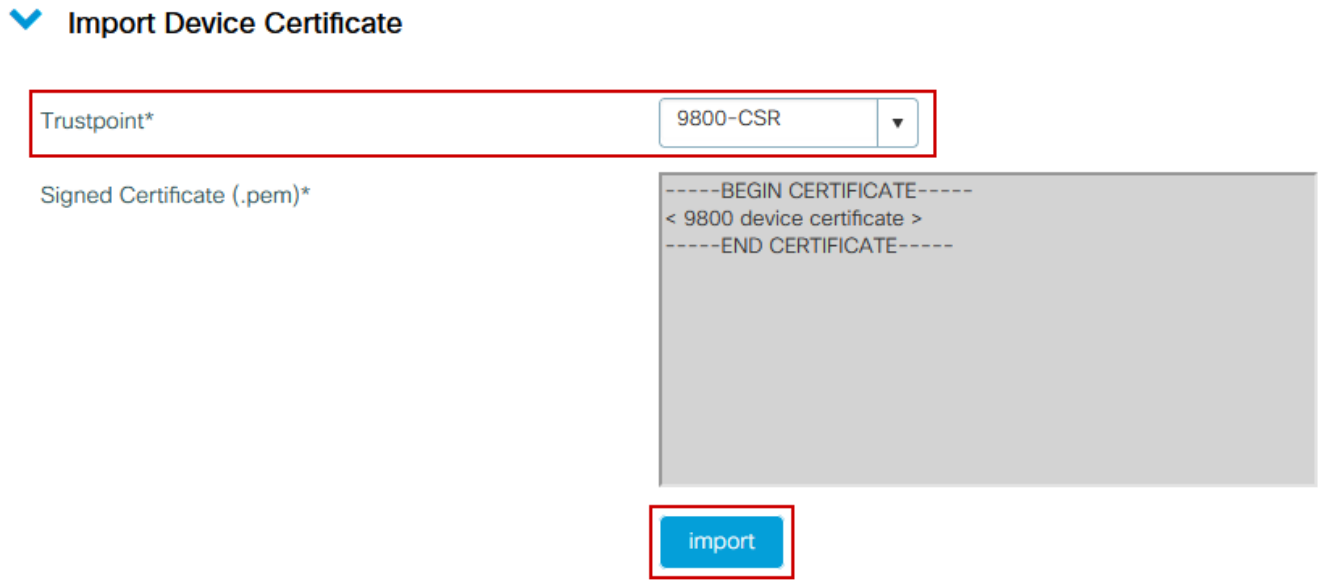
Stap 4b. In het scenario waarin meerdere machtigingsniveaus bestaan, is een nieuw trustpoint nodig voor elk CA-niveau. Deze trustpoints bevatten alleen het verificatiecertificaat en wijzen naar het volgende verificatieniveau. Dit proces wordt alleen in de CLI uitgevoerd en in dit voorbeeld zijn er één tussenliggende CA's en één root CA:

```
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
```

> **Opmerking**: als er meer dan één tussenliggende CA's in de certificeringsketen zijn, moet er een nieuw truspoint worden gegenereerd per extra certificeringsniveau. Deze truspoints moeten verwijzen naar het trustpoint dat het volgende niveau van certificatie met de bevel **ketting-bevestiging** bevat **blijven <trustpoint-name>**.

Stap 5. Laad het ondertekende certificaat in de 9800 WLC. Breid de sectie **Apparaatcertificaat importeren uit** in hetzelfde menu. Kies het eerder gedefinieerde **Trustpoint** en plak het ondertekende apparaatcertificaat dat door de CA wordt verstrekt. Klik vervolgens op **Importeren** nadat de certificaatgegevens zijn geverifieerd.



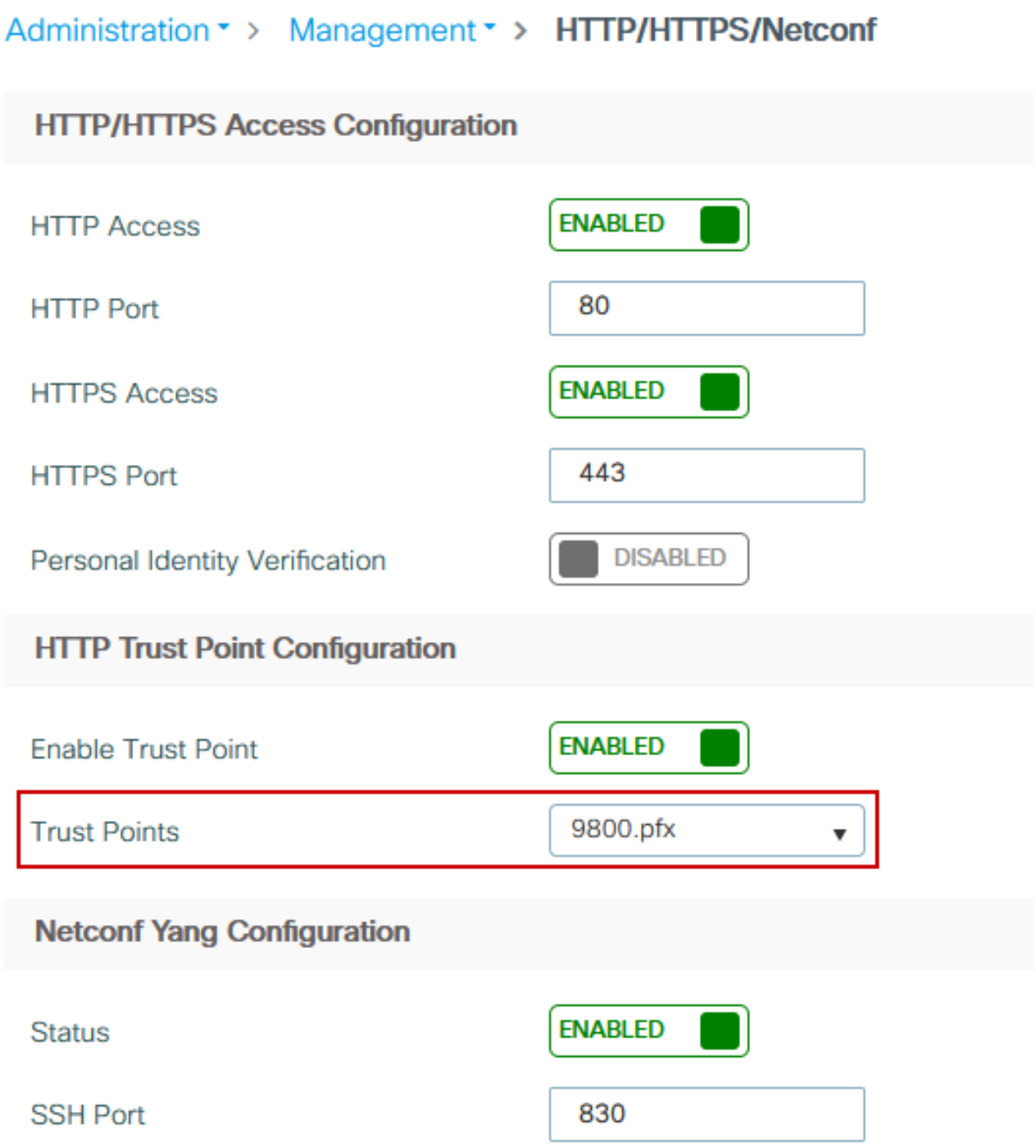
CLI-configuratie:

```
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
```

Gebruik het nieuwe certificaat

Webbeheer

Navigeer naar **Beheer > Beheer > HTTP/HTTPS/Netconf** en kies het geïmporteerde certificaat uit de vervolgkeuzelijst **Vertrouwspunten**.

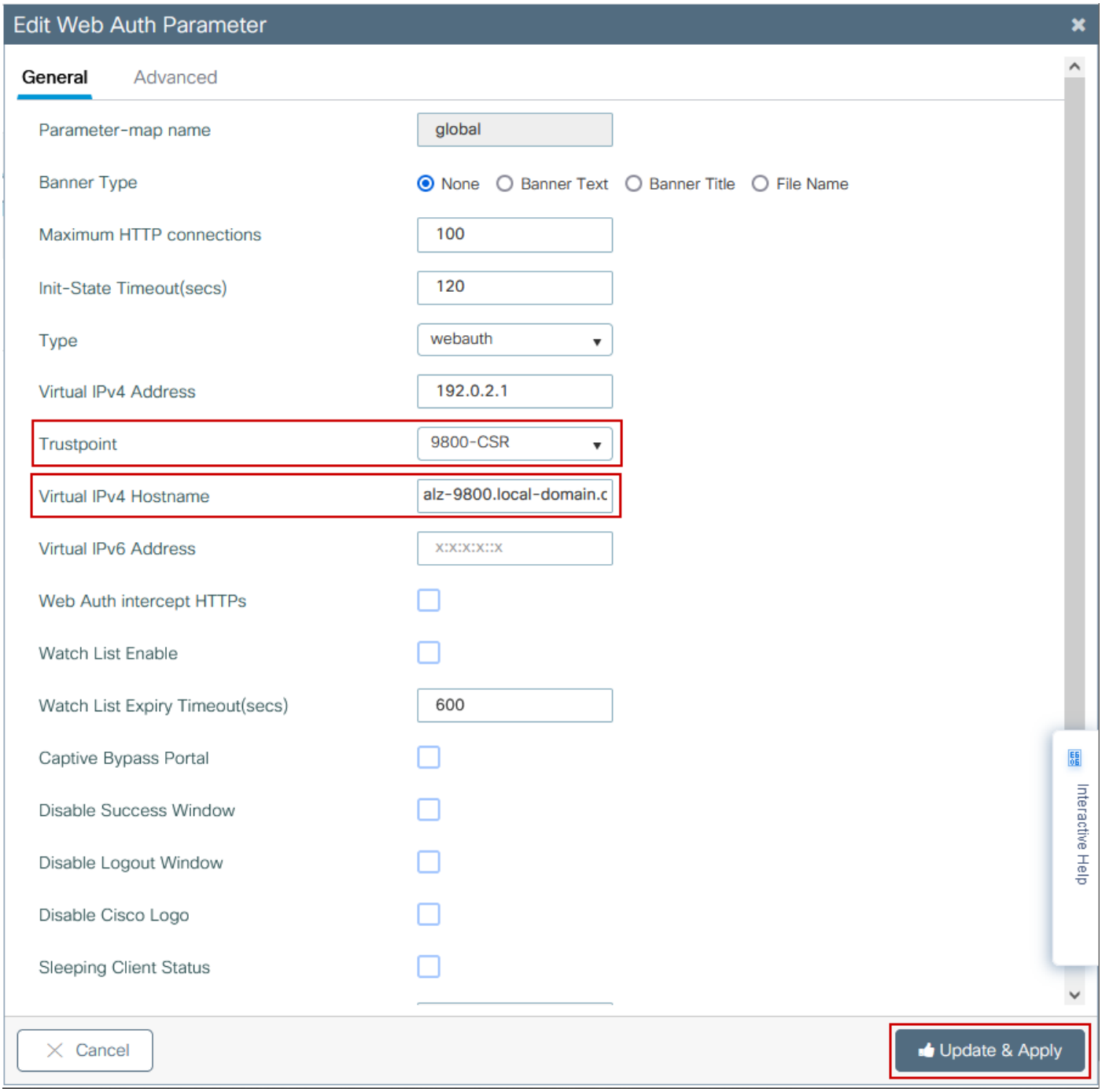


CLI-configuratie:

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

Lokale webverificatie

Navigeer naar **Configuration > Security > Web Auth**, kies de **globale** parameterkaart en kies het geïmporteerde trustpoint uit de vervolgkeuzelijst **Trustpoint**. Klik op **Bijwerken en toepassen** om de wijzigingen op te slaan. Zorg ervoor dat de **virtuele IPv4 Hostname** overeenkomt met de algemene naam in het certificaat.



CLI-configuratie:

```
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
```

Om certificaatgebruik bij te werken, moet u HTTP-services opnieuw opstarten:

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

.

Overwegingen met betrekking tot hoge beschikbaarheid

Op een 9800-paar dat is geconfigureerd voor Stateful Switchover High Availability (HA SSO), worden alle certificaten bij eerste bulk-sync gerepliceerd van het primaire naar het secundaire. Dit omvat certificaten waar de privé sleutel op het controlemechanisme zelf werd geproduceerd, zelfs als de sleutel van RSA aan niet-uitvoerbaar wordt gevormd. Nadat het HA-paar is ingesteld, wordt elk nieuw certificaat geïnstalleerd op beide controllers en worden alle certificaten in real time gerepliceerd.

Na mislukking, gebruikt de vroegere-secundair-nu-actieve controlemechanisme de certificaten die van primair doorzichtig worden geërfd.

Hoe te verzekeren het Certificaat door Webbrowsers wordt vertrouwd

Er zijn enkele belangrijke overwegingen om ervoor te zorgen dat een certificaat wordt vertrouwd door webbrowsers:

- De algemene naam (of een SAN-veld) moet overeenkomen met de URL die door de browser wordt bezocht.
- Zij moet binnen de geldigheidstermijn ervan vallen.
- Het moet worden uitgegeven door een CA of keten van CA waarvan de wortel wordt vertrouwd door de browser. Hiervoor moet het certificaat dat door de webserver wordt verstrekt alle certificaten van de keten bevatten tot (niet noodzakelijk inbegrepen) een certificaat dat door de clientbrowser wordt vertrouwd (meestal de root-CA).
- Als het herroepingslijsten bevat, moet de browser hen kunnen downloaden en het certificaat CN moet niet worden vermeld.

Verifiëren

U kunt deze opdrachten gebruiken om de configuratie van certificaten te controleren:

```
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
```

U kunt uw certificaatketting op de 9800 verifiëren. In het geval van een apparaatcertificaat dat is afgegeven door een tussenliggende CA, zelf afgegeven door een wortel CA, hebt u één vertrouwenspunt per groep van twee certificaten, zodat elk niveau zijn eigen vertrouwenspunt heeft. In dit geval heeft de 9800 WLC **9800.pfx**, met het device certificate (WLC-certificaat) en de CA (intermediaire CA) die het heeft afgegeven. Vervolgens een ander vertrouwenspunt met de Root CA die dat tussenliggende CA heeft uitgegeven.

```
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
```

Certificaatverificatie met OpenSSL

OpenSSL kan nuttig zijn om het certificaat zelf te verifiëren of om bepaalde conversiebewerkingen uit te voeren.

Zo geeft u een certificaat weer met OpenSSL:

```
openssl x509 -in
```

Zo geeft u de inhoud van een MVO weer:

```
openssl req -noout -text -in
```

Als u het eindcertificaat op de 9800 WLC wilt verifiëren maar iets anders wilt gebruiken dan uw browser, kan OpenSSL dit doen en u veel details geven.

```
openssl s_client -showcerts -verify 5 -connect
```

U kunt **<**wlcURL**>** vervangen door de URL van de webadmin van de 9800 of de URL van het gastportal (virtuele IP). Je kunt daar ook een IP-adres zetten. Het vertelt u welke certificaatketting wordt ontvangen maar de certificaatbevestiging kan nooit 100% correct zijn wanneer een IP adres in plaats van hostname wordt gebruikt.

Om de inhoud te bekijken en een PKCS12 (.pfx) certificaat of certificaatketen te verifiëren:

```
openssl pkcs12 -info -in
```

Hier is een voorbeeld van deze opdracht op een certificaatketen waarbij het apparaatcertificaat wordt afgegeven aan het Technical Assistance Center (TAC) door een intermediaire CA met de naam "intermediair.com", zelf afgegeven door een root-CA met de naam "root.com":

```
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-----
```

Problemen oplossen

Gebruik deze opdracht voor probleemoplossing. Als dit op een externe sessie (SSH of telnet) gebeurt, is er een terminalmonitor nodig om de uitgangen weer te geven:

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

Succesvol scenario debug uitvoer

Deze output toont de verwachte output wanneer een succesvolle certificaatinvoer op een 9800 gebeurt. Gebruik dit voor verwijzing en identificeer de mislukkingsstaat:

```
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.
```

Probeer een PKCS12-certificaat zonder CA te importeren

Als u een certificaat importeert en de fout krijgt: "CA cert is not found.", betekent dit dat uw .pfx bestand niet de hele keten bevat of dat één CA niet aanwezig is.

```
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.
```

Als u de opdracht **openssl pkcs12 -info -in <path to cert>** uitvoert en slechts één certificaat met één private key displays, betekent dit dat de CA niet aanwezig is. Als vuistregel, maakt deze opdracht idealiter een lijst van uw gehele keten van certificaten. Het is niet vereist om de bovenste wortel CA te omvatten als het al bekend is bij de client browsers.

Eén manier om dit te verhelpen is door de PKCS12 te deconstrueren in PEM en de keten goed te herbouwen. In het volgende voorbeeld, hadden we een .pfx bestand dat alleen het apparaat (WLC) certificaat en de sleutel bevatte. Het werd uitgegeven door een intermediaire CA (die niet aanwezig was in het PKCS12-bestand) die op zijn beurt werd ondertekend door een bekende root CA.

Stap 1. Exporteer de privé-sleutel.

```
openssl pkcs12 -in
```

Stap 2. Exporteer het certificaat als PEM.

```
openssl pkcs12 -in
```

Stap 3. Download het tussenliggende CA-certificaat als PEM.

De bron van CA hangt af van de aard ervan, als het een openbare CA is dan is een online zoekopdracht voldoende om de repository te vinden. Anders moet de CA-beheerder de certificaten in Base64-formaat (.pem) leveren. Als er meerdere CA-niveaus zijn, groepeert u deze in één bestand, zoals het bestand dat wordt weergegeven aan het einde van het importproces van **optie 1**.

Stap 4. Herstel de PKCS 12 vanaf de sleutel, apparaatbeveiliging en CA-certificaat.

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

We hebben nu "fixedcertchain.pfx" die we graag importeren naar Catalyst 9800!

Opmerkingen en beperkingen

- Cisco IOS® XE ondersteunt CA-certificaten niet met een geldige waarde na 2099: Cisco bug-id CSCvp64208
- Cisco IOS® XE ondersteunt de SHA256-berichtsamenvatting PKCS 12-bundel (SHA256 certs worden ondersteund, maar niet als de PKCS 12-bundel zelf is ondertekend met SHA256): Cisco bug-id CSCvz41428
- U kunt fragmentatie zien als de WLC gebruikerscertificaten moet dragen, en het NAC/ISE-apparaat is bereikbaar via het internet (bijvoorbeeld in een SD-WAN-implementatie). Certificaten zijn bijna altijd groter dan 1500 bytes (wat betekent dat er meerdere RADIUS-pakketten worden verzonden om het certificaatbericht te dragen) en als u meerdere verschillende MTU over het netwerkpad hebt, kan er meer fragmentatie van de RADIUS-pakketten zelf optreden. In zulke gevallen raden we u aan om al uw UDP datagrammen voor het WLC-verkeer via dezelfde route te versturen om problemen zoals vertraging/jitter, die kunnen worden veroorzaakt door het weer op internet, te voorkomen

Over deze vertaling

Cisco heeft dit document vertaald via een combinatie van machine- en menselijke technologie om onze gebruikers wereldwijd ondersteuningscontent te bieden in hun eigen taal. Houd er rekening mee dat zelfs de beste machinevertaling niet net zo nauwkeurig is als die van een professionele vertaler. Cisco Systems, Inc. is niet aansprakelijk voor de nauwkeurigheid van deze vertalingen en raadt aan altijd het oorspronkelijke Engelstalige document (link) te raadplegen.