تشكيل RADIUS DTLS على ISE و 9800 WLC

المحتويات

[المقدمة](#)
[الخلفية](#)
[المتطلبات الأساسية](#)
[المتطلبات](#)
[المكونات المستخدمة](#)
[التكوين](#)
[نظرة عامة](#)
[اختياري - إنشاء شهادة جهاز WLC و ISE RADIUS DTLS](#)
[إضافة مقاطع التكوين على ملف openssl.cnf](#)
[إنشاء شهادة جهاز WLC](#)
[إنشاء شهادة جهاز ISE](#)
[إستيراد الشهادات إلى الأجهزة](#)
[إستيراد الشهادات إلى ISE](#)
[إستيراد الشهادات إلى WLC](#)
[تكوين RADIUS DTLS](#)
[تكوين ISE](#)
[تكوين وحدة التحكّم في شبكة LAN اللاسلكية (WLC)](#)
[التحقق من الصحة](#)
[التحقق من معلومات الشهادة](#)
[إجراء إختبار المصادقة](#)
[استكشاف الأخطاء وإصلاحها](#)
[CA غير معروف تم الإعلام عنه بواسطة WLC](#)
[CA غير معروف تم الإعلام عنه بواسطة ISE](#)
[التحقق من البطاقة موجود في المكان](#)
[أستكشاف أخطاء إنشاء نفق DTLS وإصلاحها على التقاط الحزمة](#)

المقدمة

يصف هذا وثيقة طريقة أن يخلق الترخيص ضروري أن يشكل RADIUS DTLS بين ISE و 9800 WLC.

الخلفية

RADIUS DTLS هو شكل آمن لبروتوكول RADIUS حيث يتم إرسال رسائل RADIUS عبر نفق أمان طبقة نقل البيانات (DTLS). لإنشاء هذا النفق بين خادم المصادقة والمصادقة، يلزم مجموعة من الشهادات. تتطلب هذه المجموعة من الشهادات تعيين امتدادات شهادات إستخدام المفتاح الموسع (EKU)، وعلى وجه التحديد، مصادقة العميل على شهادة WLC ومصادقة الخادم بالإضافة إلى مصادقة العميل لشهادة ISE.

المتطلبات الأساسية

المتطلبات

توصي Cisco بأن تكون لديك معرفة بالمواضيع التالية:

- كيفية تكوين عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) طراز 9800، نقطة الوصول (AP) للتشغيل الأساسي
- كيفية إستخدام تطبيق OpenSSL
- البنية الأساسية للمفتاح العام (PKI) والشهادات الرقمية

المكونات المستخدمة

تستند المعلومات الواردة في هذا المستند إلى إصدارات البرامج والمكونات المادية التالية:

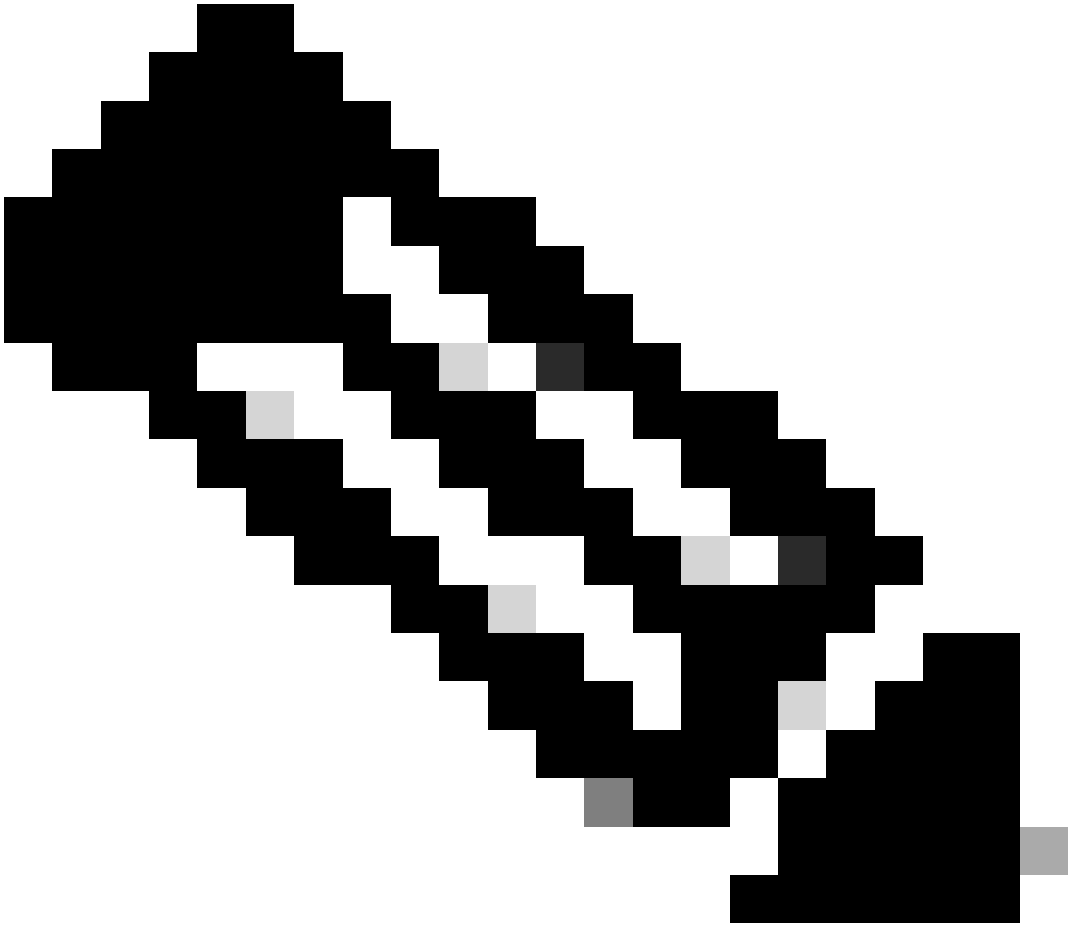
- تطبيق OpenSSL (الإصدار 3.0.2).
- ISE (الإصدار 3.1.0.518)
- 9800 WLC (الإصدار 17.12.3)

تم إنشاء المعلومات الواردة في هذا المستند من الأجهزة الموجودة في بيئة معملية خاصة. بدأت جميع الأجهزة المُستخدمة في هذا المستند بتكوين ممسوح (افتراضي). إذا كانت شبكتك قيد التشغيل، فتأكد من فهمك للتأثير المحتمل لأي أمر.

التكوين

نظرة عامة

الغرض هو إنشاء مرجع مصدق متوسط و مرجع مصدق جذر ثنائي المستوى مع مرجع مصدق ثانوي لتوقيع شهادات نقطة النهاية. بمجرد توقيع الشهادات، يتم إستيرادها إلى عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) وISE. أخيرا، يتم تكوين الأجهزة لإجراء مصادقة RADIUS DTLS باستخدام هذه الشهادات.



ملاحظة: يستخدم هذا المستند أوامر Linux المحددة لإنشاء الملفات وترتيبها. يتم شرح الأوامر بحيث يمكنك تنفيذ الإجراء نفسه على أنظمة التشغيل الأخرى حيث يتوفر OpenSSL.

إختياري - إنشاء شهادة جهاز WLC و ISE RADIUS DTLS

يحتاج بروتوكول RADIUS DTLS إلى تبادل الشهادات بين ISE و WLC لإنشاء نفق DTLS. إذا لم تكن لديك شهادات صالحة بعد، يمكنك إنشاء مرجع مصدق محلي لإنشاء الشهادات، ارجع إلى [تكوين مرجع شهادات متعدد المستويات على OpenSSL لإنشاء شهادات متوافقة مع Cisco IOS® XE](#) وقم بتنفيذ الخطوات المحددة على المستند من البداية حتى نهاية الخطوة إنشاء شهادة CA وسيطة.

إضافة مقاطع التكوين على ملف openssl.cnf

افتح ملف تكوين openssl.cnf الخاص بك، وفي أسفله، انسخ والصق أقسام WLC و ISE المستخدمة لإنشاء طلب توقيع شهادة صالح (CSR).

ةقطنم تاكبشب ةمئاق ىلإ WLC_device_req_extو ISE_device_req_ext مسق نم لك ريشي
نيزختلا (SAN) يتلا متيس اهنيمضت ىلع CSR:

```
#Section used for CSR generation, it points to the list of subject alternative names to add them to CSR
[ ISE_device_req_ext ]
subjectAltName = @ISE_alt_names

[ WLC_device_req_ext ]
subjectAltName = @WLC_alt_names

#DEFINE HERE SANS/IPs NEEDED for **ISE** device certificates
[ISE_alt_names]
DNS.1   = ISE.example.com
DNS.2   = ISE2.example.com

#DEFINE HERE SANS/IPs NEEDED for **WLC** device certificates
[WLC_alt_names]
DNS.1   = WLC.example.com
DNS.2   = WLC2.example.com
```

ال ثيحب اهعيقوتل CSR ىلع ةدوجوم (SAN) نيزخت تاكبش يأ CA ىطختي ،ينمأ ريبدتك
لجأ نم .اهمادختساب حومسم ريغ مسالا ةحلاص ةداهش يقلتت اهب حرصملا ريغ ةزهجألل نكمي
ةفاضإ تاكبش ةقطنم نيزختلا (SANs) ةرم ىرخأ ىلإ ةداهشلا ةعقوملا، مدختسأ ةملعملا
subjectAltName ةراشإلل ىلإ سفن مئاوق تاكبش ةقطنم نيزختلا (SAN) ةمدختسملاك
ءاشنإل CSR.

بلطتي ISE دوجو EKU نم ServerAuth وClientAuth ىلع ةداهشلا امنيب جاتحي WLC طقف ىلإ
ClientAuth. متت اهتفاضإ ىلإ ةداهشلا ةعقوملا مادختساب ةملعملا extendedKeyUsage.

خسنا عطاقملا ةمدختسملا عيقوتل ةداهشلا اهقصلو يف لفسأ فلم openssl.cnf:

```
#This section contains the extensions used for the device certificate sign
[ ISE_cert ]
basicConstraints=CA:FALSE
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer:always
#EKU client and server is needed for RADIUS DTLS on ISE
extendedKeyUsage = serverAuth, clientAuth
subjectAltName = @ISE_alt_names

[ WLC_cert ]
basicConstraints=CA:FALSE
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer:always
#EKU client is needed for RADIUS DTLS on WLC
extendedKeyUsage = clientAuth
subjectAltName = @WLC_alt_names
```

إنشاء شهادة جهاز WLC

قم بإنشاء مجلد جديد لتخزين قوائم التحكم في الشبكة المحلية اللاسلكية (WLC) على الجهاز الذي تم تثبيت OpenSSL عليه داخل مجلد شهادة CA الوسيط المسمى IntermCA.db.certs. يسمى المجلد الجديد WLC:

```
mkdir ./IntermCA/IntermCA.db.certs/WLC
```

قم بتعديل معلمات DNS في قسم [WLC_alt_names] في ملف openssl.cnf. قم بتغيير أسماء المثلة المقدمة لقيمك المطلوبة. تقوم هذه القيم بملء حقل SANs من شهادة WLC:

```
[WLC_alt_names]
DNS.1   = WLC.example.com    <-----Change the values after the equals sign
DNS.2   = WLC2.example.com   <-----Change the values after the equals sign
```

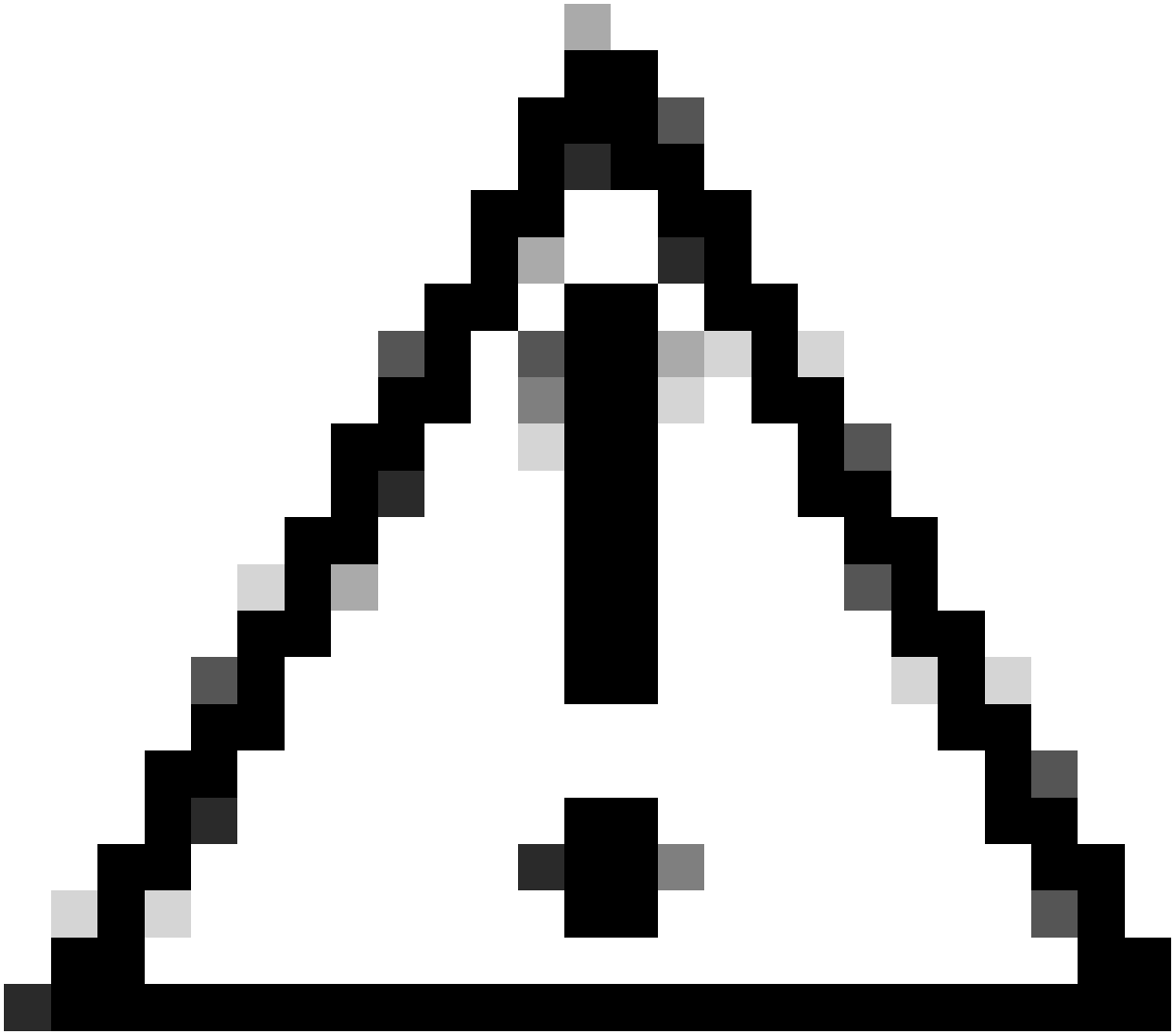
قم بإنشاء مفتاح WLC الخاص و WLC CSR باستخدام معلومات من القسم WLC_device_req_ext لشبكات منطقة التخزين (SANs):

```
openssl req -newkey rsa:4096 -keyout ./IntermCA/IntermCA.db.certs/WLC/WLC.key -nodes -config openssl.cn
```

في حت OpenSSL رسالة خ تفاعلية لإدخال تفاصيل الاسم المميز (DN):

```
.........+..+.......+..........+...+......+.+...........+..+++++++++++++
++++++
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name [MX]:
State or province [CDMX]:
Locality [CDMX]:
Organization name [Cisco lab]:
Organizational unit [Cisco Wireless]:
Common name []:WLC.example.com
```

المطابقة التفاعلية لاسم شهادة WLC المميز



تحذير: يجب أن يكون الاسم الشائع (CN) الذي توفره في موجه الأمر التفاعلي مطابقا لأحد الأسماء الموجودة في قسم [WLC_alt_names] في ملف openssl.cnf.

أستخدم المرجع المصدق المسمى IntermCA لتوقيع WLC CSR المسمى WLC.csr مع الملحقات المحددة ضمن [WLC_CERT] وتخزين الشهادة الموقعة داخل ./IntermCA/IntermCA.db.certs/WLC. تسمى شهادة جهاز WLC WLC.crt:

```
openssl ca -config openssl.cnf -extensions WLC_cert -name IntermCA -out ./IntermCA/IntermCA.db.certs/WL
```

يحتاج 9800 WLC أن تكون الشهادة بتنسيق PFX لاستيرادها. إنشاء ملف جديد يحتوي على سلسلة CAs التي وقعت على شهادة WLC، يسمى هذا ملف certfile:

```
cat ./RootCA/RootCA.crt ./IntermCA/IntermCA.crt > ./IntermCA/IntermCA.db.certs/WLC/certfile.crt
```

إلى ملف .pfx، قم بتشغيل أحد هذه الأوامر وفقا لإصدار WLC.

للإصدارات الأقدم من 17.12.1:

```
openssl pkcs12 -export -macalg sha1 -legacy -descert -out ./IntermCA/IntermCA.db.certs/WLC/WLC.pfx -ink
```

للإصدار 17.12.1 وأحدث:

```
openssl pkcs12 -export -out ./IntermCA/IntermCA.db.certs/WLC/WLC.pfx -inkey ./IntermCA/IntermCA.db.certs
```

إنشاء شهادة جهاز ISE

قم بإنشاء مجلد جديد لتخزين بيانات ISE على الجهاز الذي تم تثبيت OpenSSL عليه داخل مجلد شهادة CA الوسيطة المسمى IntermCA.db.certs. يسمى المجلد الجديد ISE:

```
mkdir ./IntermCA/IntermCA.db.certs/ISE
```

قم بتعديل معلمات DNS في قسم [ISE_alt_names] في ملف openssl.cnf. تغيير أسماء الأمثلة المقدمة للقيم المرغوبة، وقم بملء حقل SANs من شهادة WLC:

```
[ISE_alt_names]
DNS.1   = ISE.example.com   <-----Change the values after the equals sign
DNS.2   = ISE2.example.com  <-----Change the values after the equals sign
```

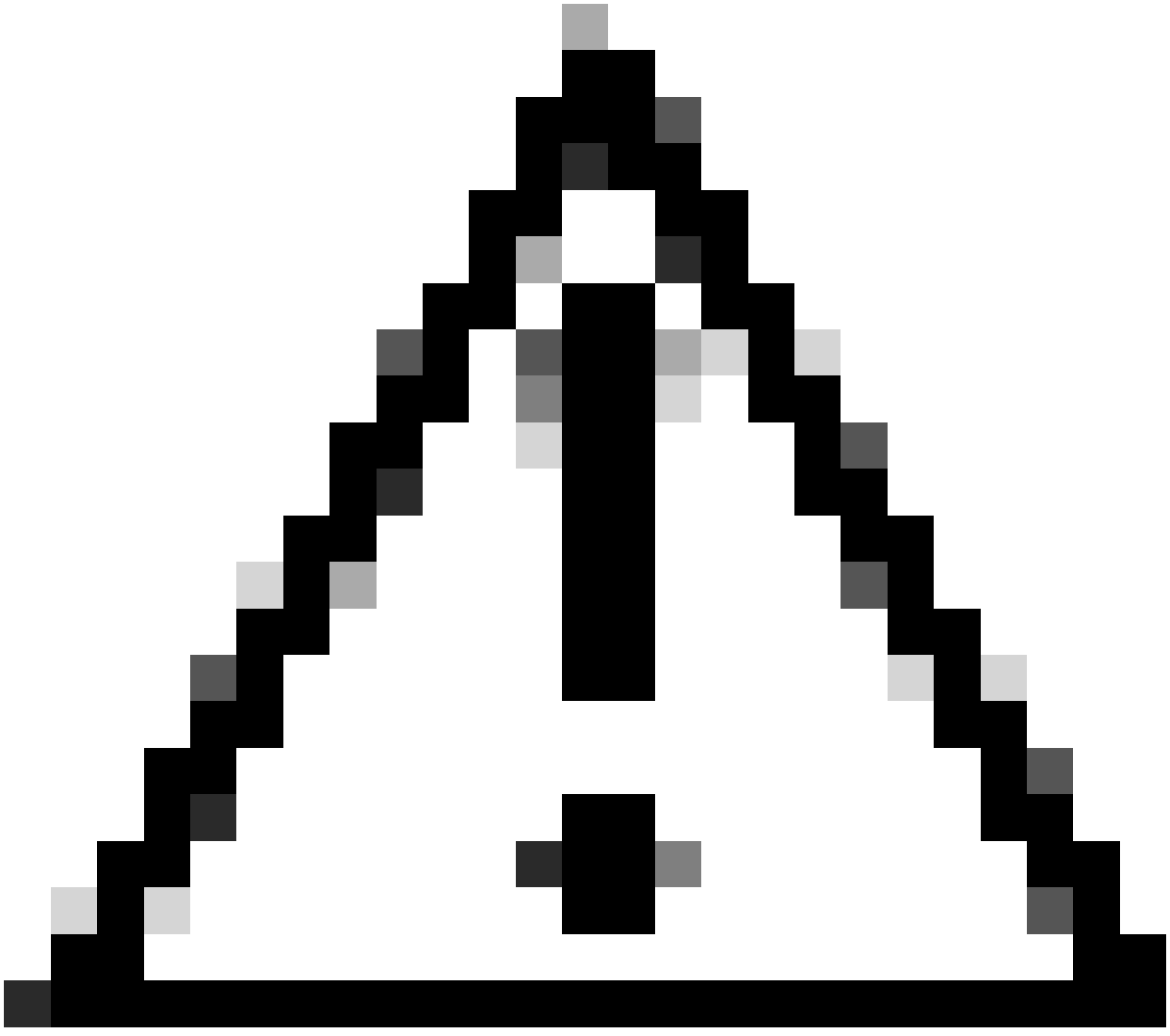
قم بإنشاء مفتاح ISE الخاص و ISE CSR باستخدام معلومات من القسم ISE_DEVICE_req_ext لتخزين (SANs) لشبكات متعددة:

```
openssl req -newkey rsa:2048 -sha256 -keyout ./IntermCA/IntermCA.db.certs/ISE/ISE.key -nodes -config op
```

يفتح OpenSSL رسالة حث تفاعلية لإدخال تفاصيل الاسم المميز (DN):

```
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name [MX]:
State or province [CDMX]:
Locality [CDMX]:
Organization name [Cisco lab]:
Organizational unit [Cisco Wireless]:
Common name []:ISE.example.com
```

المطالبة التفاعلية للاسم المميز لشهادة ISE



تحذير: يجب أن يكون CN الذي توفره على موجه الأمر التفاعلي مماثلا تماما لما هو

موجود في قسم [ISE_ALT_NAMES] في ملف openssl.cnf.

أستخدم CA المسمى IntermCA لتوقيع ISE CSR المسمى ISE.csr مع الملحقات المحددة ضمن [ISE_CERT] وتخزين الشهادة الموقعة داخل ./IntermCA/IntermCA.db.certs/WLC. تسمى شهادة جهاز ISE ISE.CRT:

```
openssl ca -config openssl.cnf -extensions ISE_cert -name IntermCA -out ./IntermCA/IntermCA.db.certs/IS
```

إستيراد الشهادات إلى الأجهزة

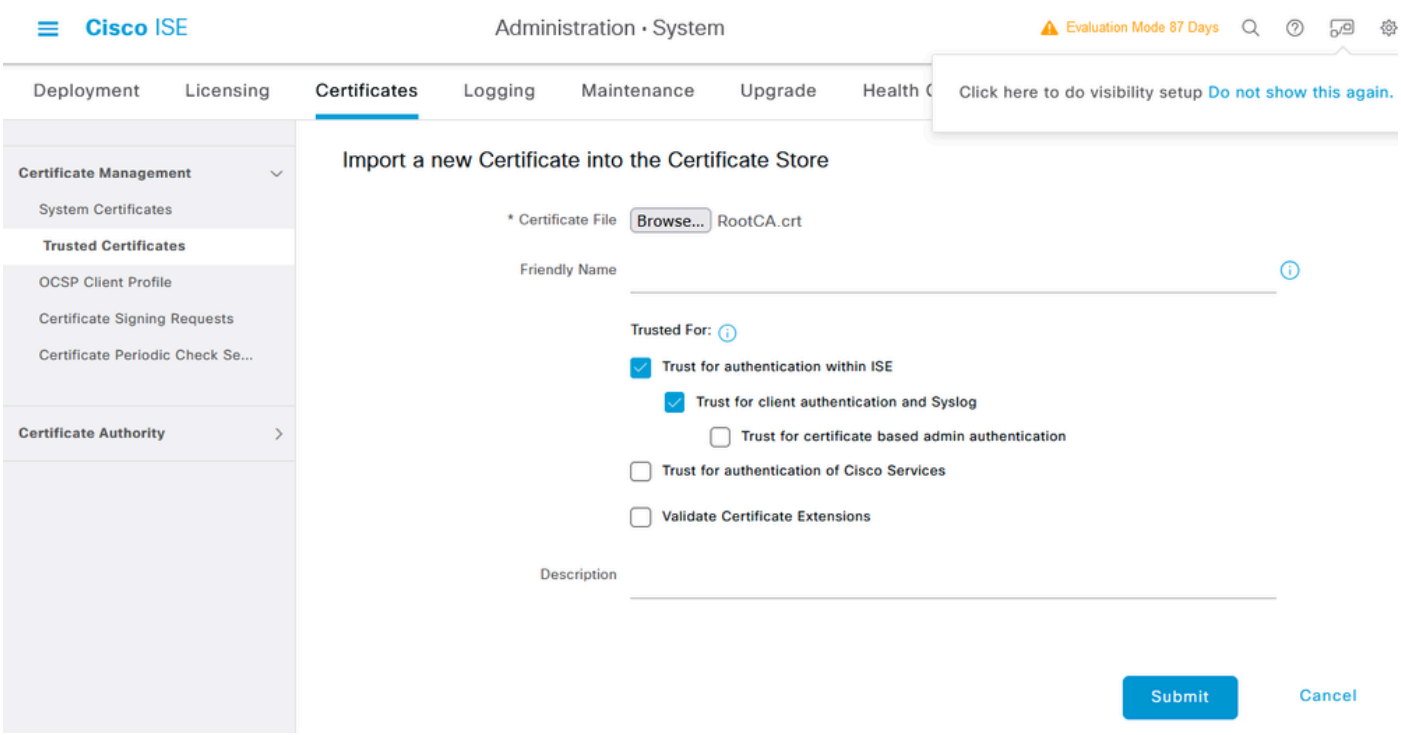
إستيراد الشهادات إلى ISE

1. إستيراد شهادة المرجع المصدق الجذر من سلسلة شهادة ISE إلى مخزن الشهادات الموثوق به.

2. انتقل إلى Administration>System>Certificates>Trusted Certificates.

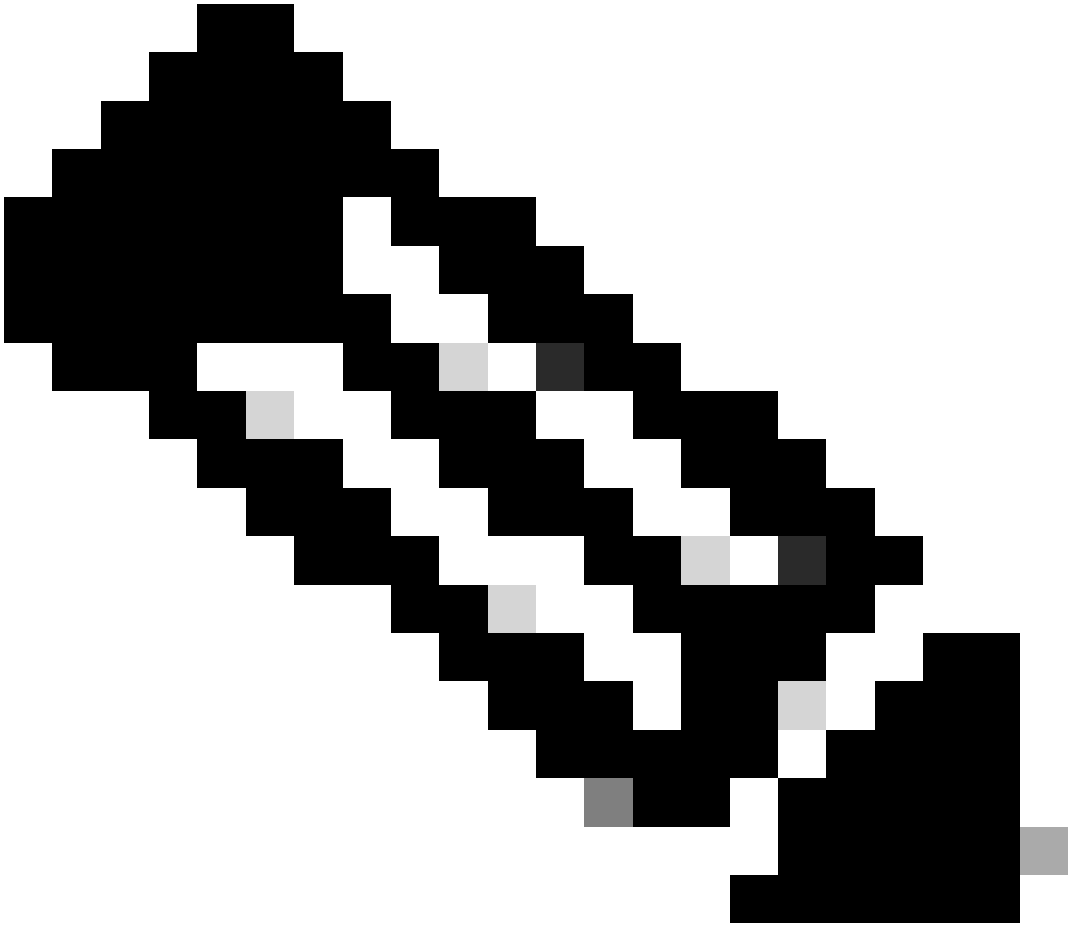
3. انقر فوق إستعراض وحدد الملف root.crt.

4. حدد خانات الاختيار الثقة للمصادقة داخل ISE بالإضافة إلى الثقة لمصادقة العميل و Syslog ثم انقر فوق إرسال:

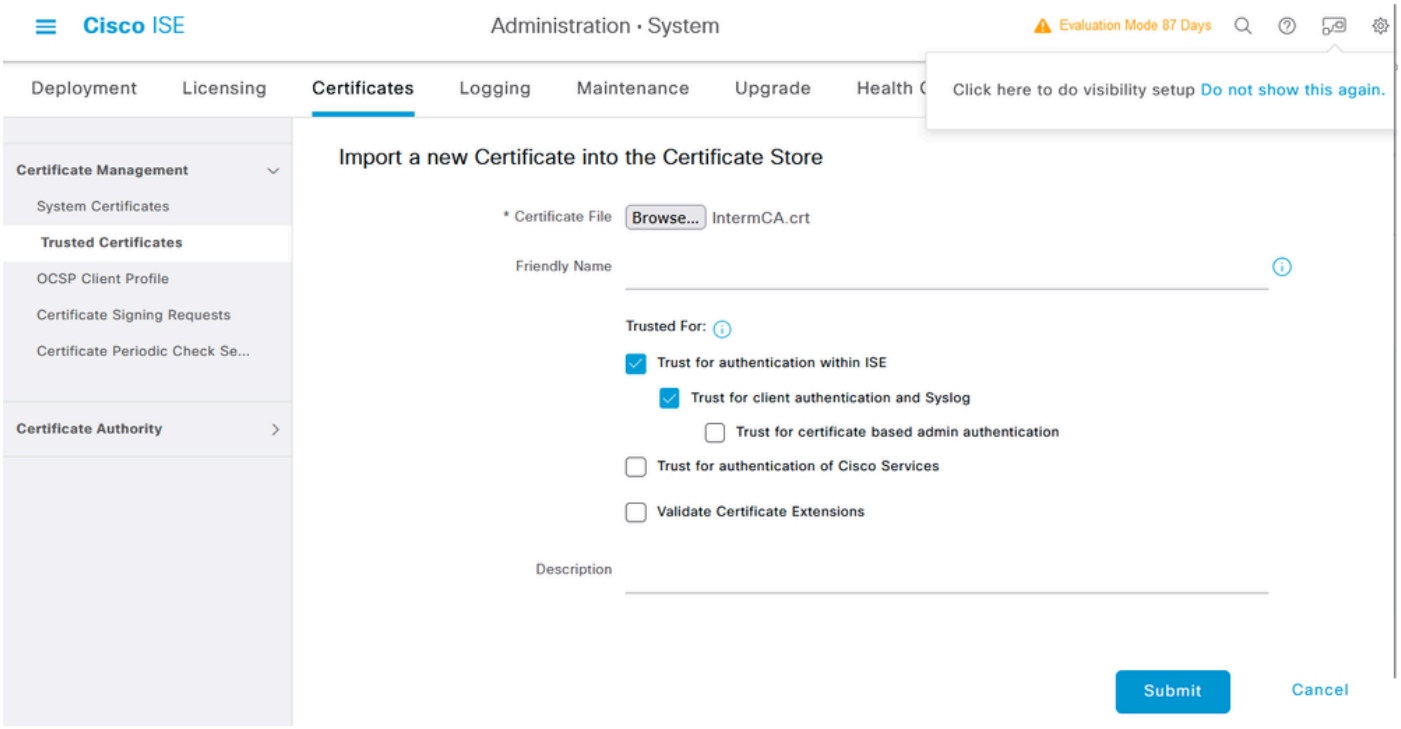


مربع حوار إستيراد شهادة ISE الجذر

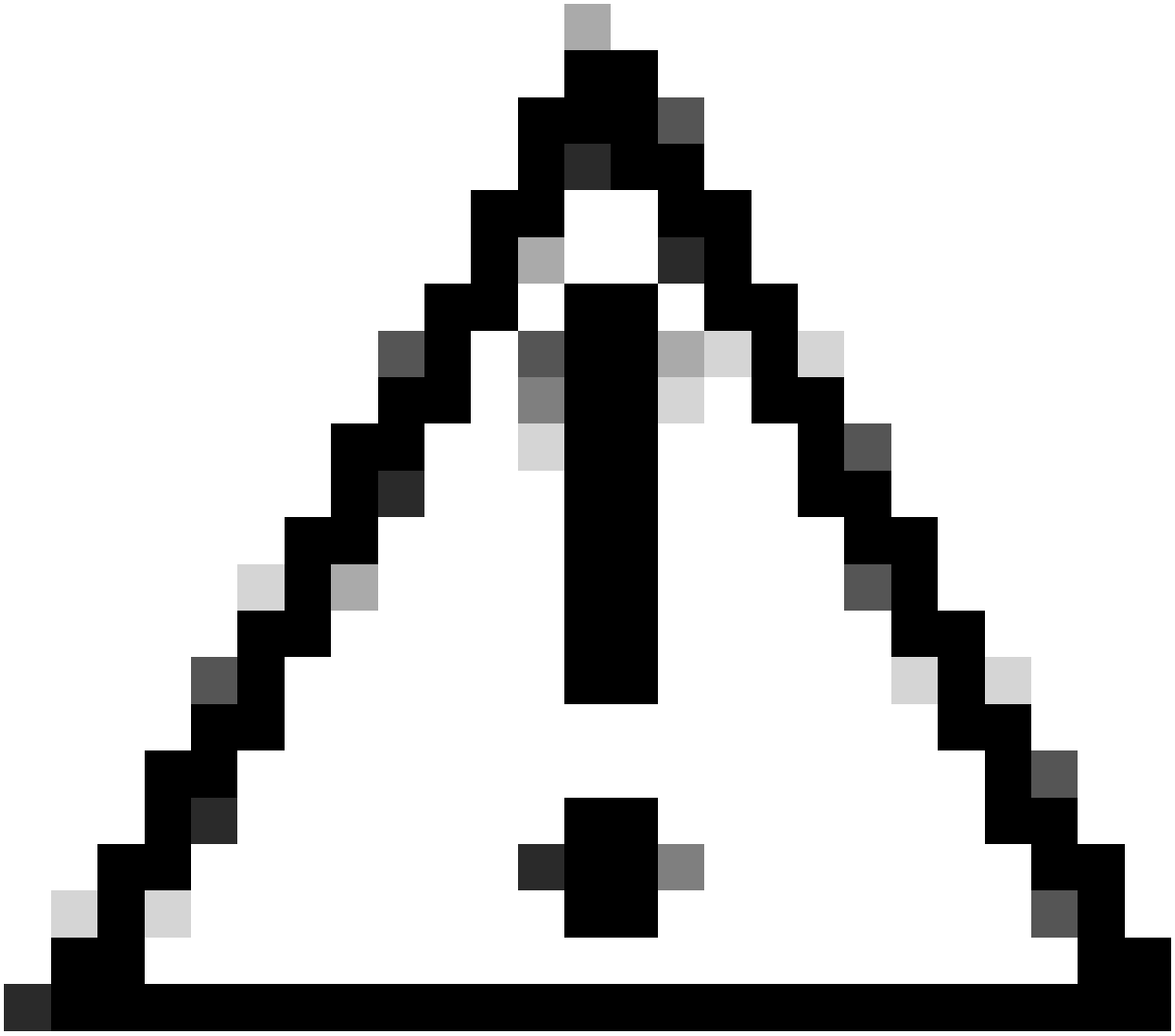
قم بالمثل للشهادة الوسيطة إذا كانت موجودة.



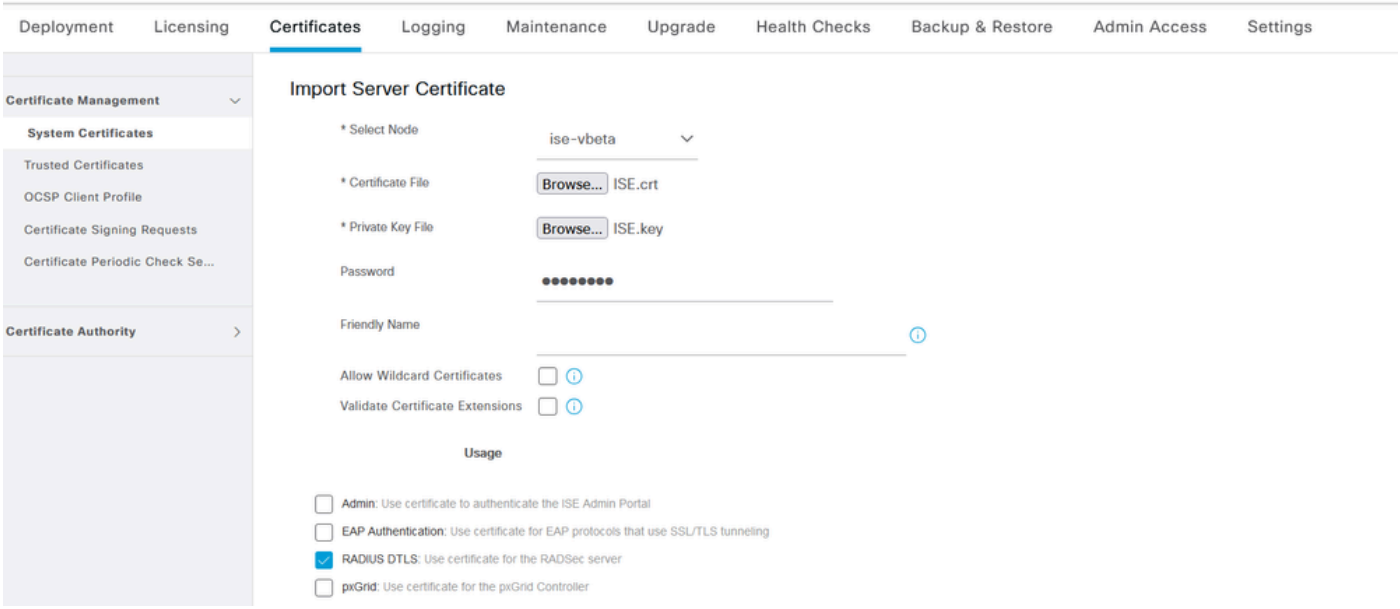
ملاحظة: كرر الخطوات الخاصة بأي شهادة شهادة CA تشكل جزءا من سلسلة التحقق من صحة شهادة ISE. ابدأ دائما بشهادة CA الجذر وتنتهي بأقل شهادة CA وسيطة للسلسلة.



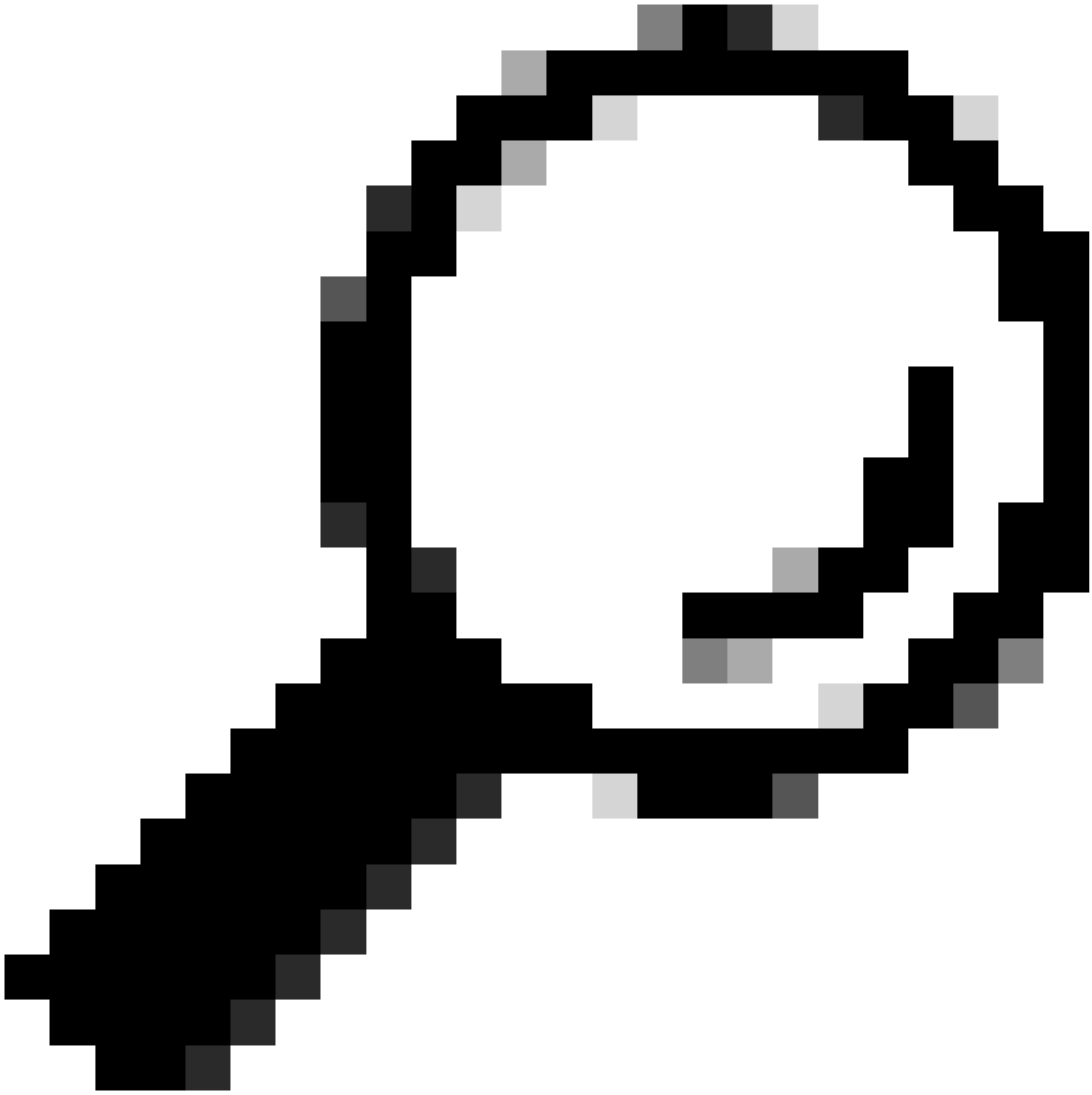
مربع حوار استيراد شهادة ISE الوسيطة CA



تحذير: إذا تم إصدار شهادة ISE وشهادة WLC من قبل CAs مختلفة، فيجب عليك إستيراد جميع شهادات CA التي تنتمي إلى سلسلة شهادات WLC كذلك. ال يقبل ISE شهادة WLC في تبادل شهادات DTLS حتى تقوم باستيراد شهادات CA هذه.



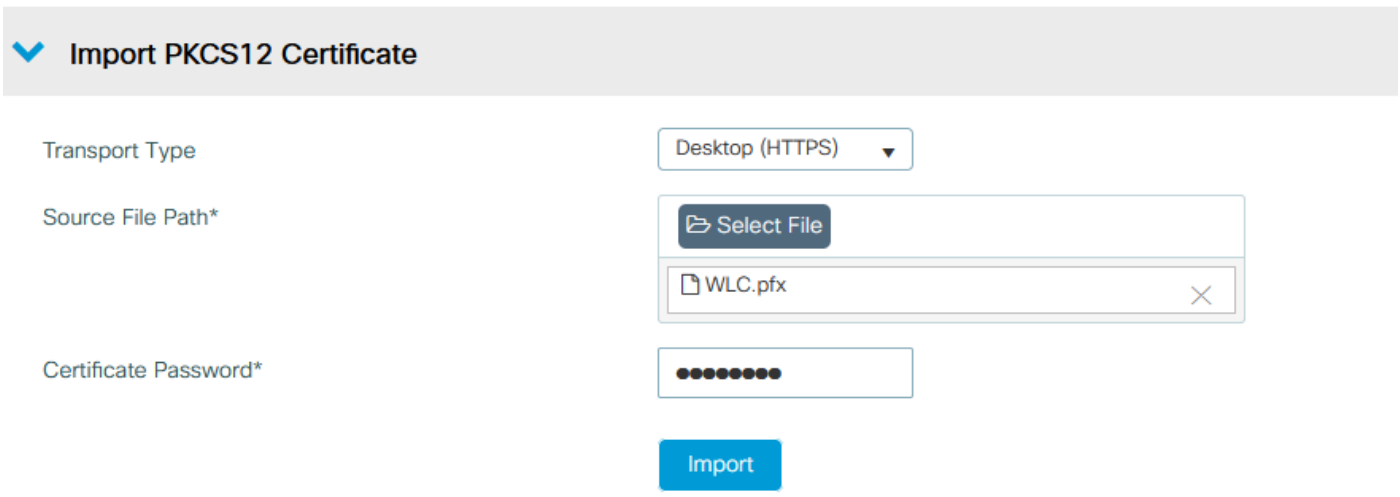
قائمة إستيراد شهادة جهاز ISE



تلميح: تحتاج فقط إلى إستيراد شهادة جهاز ISE في هذه الخطوة. هذه الشهادة هي إحدى عمليات تبادل ISE لإنشاء نفق DTLS. ليس من الضروري إستيراد شهادة جهاز WLC والمفتاح الخاص لأنه يتم التحقق من شهادة WLC باستخدام شهادات CA التي تم إستيرادها مسبقا.

إستيراد الشهادات إلى WLC

1. انتقل إلى التكوين > الأمان > إدارة PKI على عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) وانتقل إلى عالمة التبويب إضافة شهادة.
2. انقر على القائمة المنسدلة إستيراد شهادة PKCS12 وقم بضبط نوع النقل على أنه سطح مكتب (HTTPS).
3. انقر فوق الزر تحديد ملف وحدد ملف .pfx الذي قمت بإعداده مسبقا.
4. اكتب كلمة مرور الاستيراد وانقر أخيرا على إستيراد.

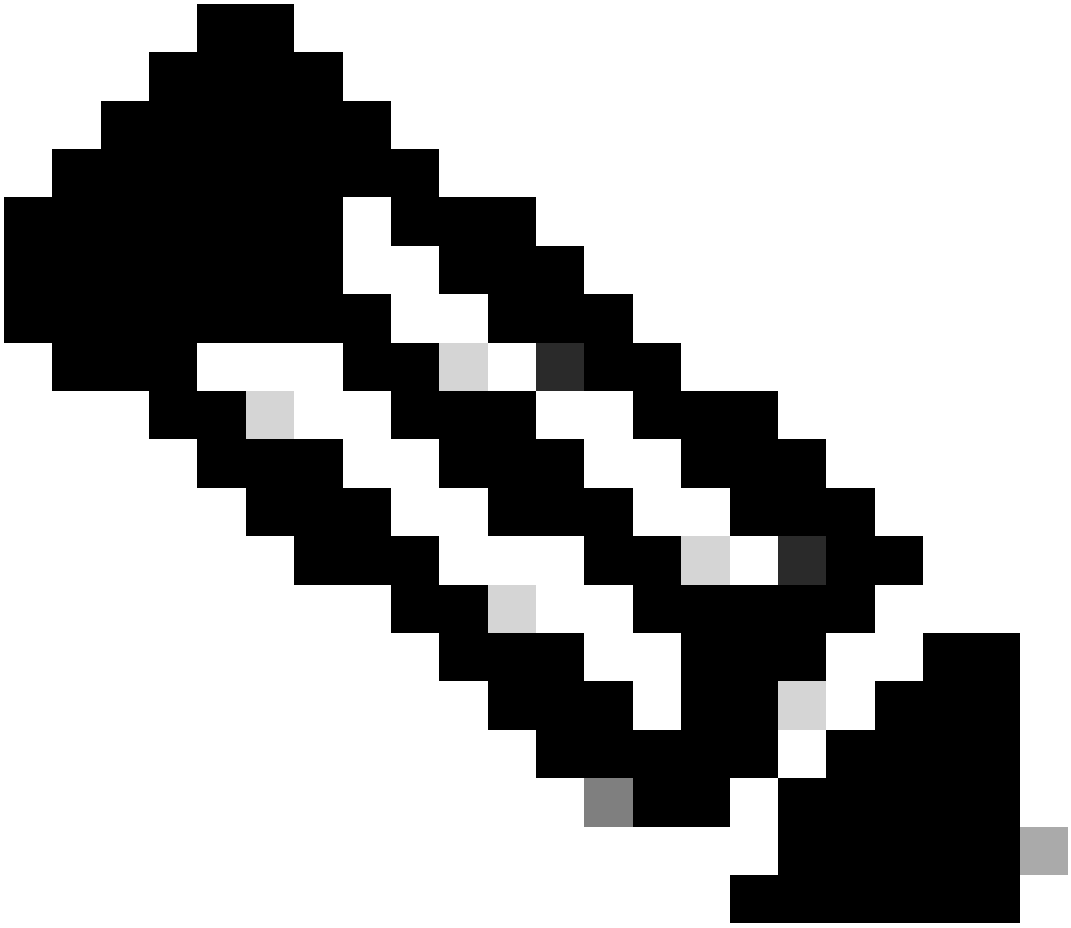


مربع حوار إستيراد شهادة WLC

للحصول على معلومات تفصيلية حول عملية الاستيراد ارجع إلى [إنشاء وتنزيل شهادات CSR على Catalyst 9800 WLCs.]

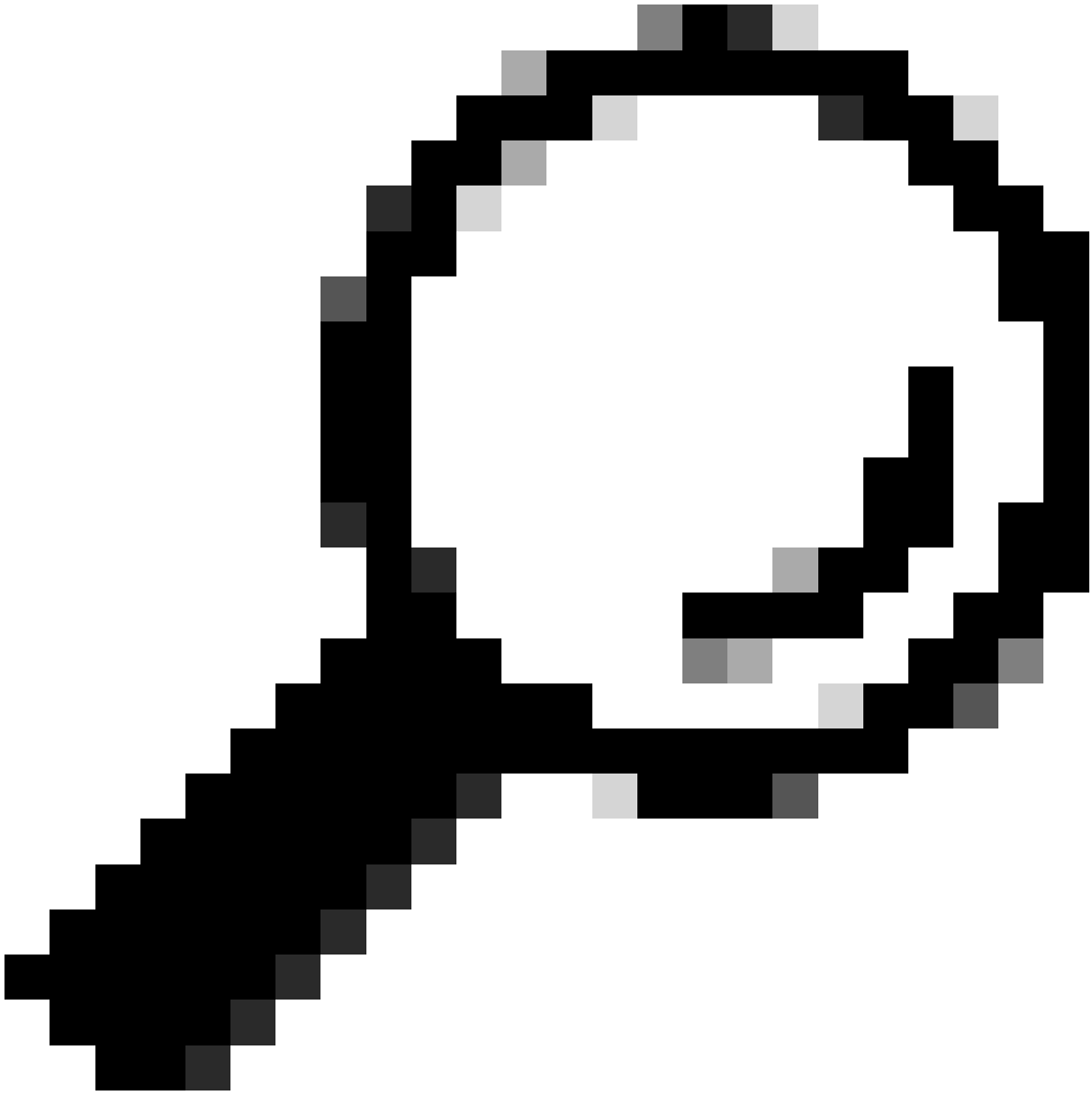
قم بتعطيل التحقق من الإبطال داخل كل نقطة ثقة تم إنشاؤها تلقائيا إذا لم يكن لدى عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) قائمة إبطال الشهادات التي يمكن التحقق منها من خلال الشبكة:

```
9800#configure terminal

9800(config)#crypto pki trustpoint WLC.pfx
9800(config)#revocation-check none
9800(config)#exit

9800(config)#crypto pki trustpoint WLC.pfx-rrr1
9800(config)#revocation-check none
9800(config)#exit
```



ملاحظة: إذا قمت بإنشاء CA متعدد المستويات على OpenSSL باستخدام تكوين CA متعدد المستويات على OpenSSL لإنشاء مستند شهادات Cisco IOS XE، فيجب عليك تعطيل التحقق من الإبطال لعدم إنشاء خادم CRL.

يقوم الاستيراد التلقائي بإنشاء نقاط الثقة اللازمة لاحتواء شهادة WLC وشهادات CA الخاصة بها.



تلميح: إذا تم إصدار شهادات WLC من نفس CA مثل شهادات ISE، يمكنك إستخدام نفس نقاط الثقة التي تم إنشاؤها تلقائيا من إستيراد شهادة WLC. لا توجد حاجة لاستيراد شهادات ISE بشكل منفصل.

إذا تم إصدار شهادة WLC من مرجع مصدق مختلف عن شهادة ISE، فأنت تحتاج أيضا إلى إستيراد شهادات ISE CA إلى عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) لكي تثق في شهادة جهاز ISE.
إنشاء نقطة ثقة جديدة ل CA الجذر واستيراد ISE Root CA:

```
9800(config)#crypto pki trustpoint ISEroot
9800(ca-trustpoint)#revocation-check none
```

```
9800(ca-trustpoint)#enrollment terminal
9800(ca-trustpoint)#chain-validation stop
9800(ca-trustpoint)#exit
9800(config)#crypto pki authenticate ISEroot

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

------Paste the ISE root CA-----
```

إستيراد الشهادة CA الوسيطة التالية على سلسلة ISE CA، بعبارة أخرى، شهادة CA الصادرة عن المرجع المصدق الجذر:

```
hamariomed1(config)#crypto pki trustpoint ISEintermediate
hamariomed1(ca-trustpoint)#revocation-check none
hamariomed1(ca-trustpoint)#chain-validation continue ISErootCA
hamariomed1(ca-trustpoint)#enrollment terminal
hamariomed1(ca-trustpoint)#exit

hamariomed1(config)#crypto pki authenticate ISEintermediate

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

------Paste the ISE intermediate CA-------
```

يتطلب كل مرجع مصدق إضافي في السلسلة وجود نقطة ثقة منفصلة. يجب أن تشير الشهادة المصدر للشهادة إلى TrustPoint التي تحتوي على شهادة المصدر للشهادة في السلسلة إلى نقطة ثقة في السلسلة التي تريد إستيرادها مع متابعة التحقق من صحة سلسلة الأوامر <Issuer TrustPoint name>.

قم باستيراد نفس عدد شهادات المرجع المصدق التي تحتوي عليها سلسلة المرجع المصدق. يتم الانتهاء بعد إستيراد المرجع المصدر من شهادة جهاز ISE، مع ملاحظة اسم نقطة الثقة هذه.

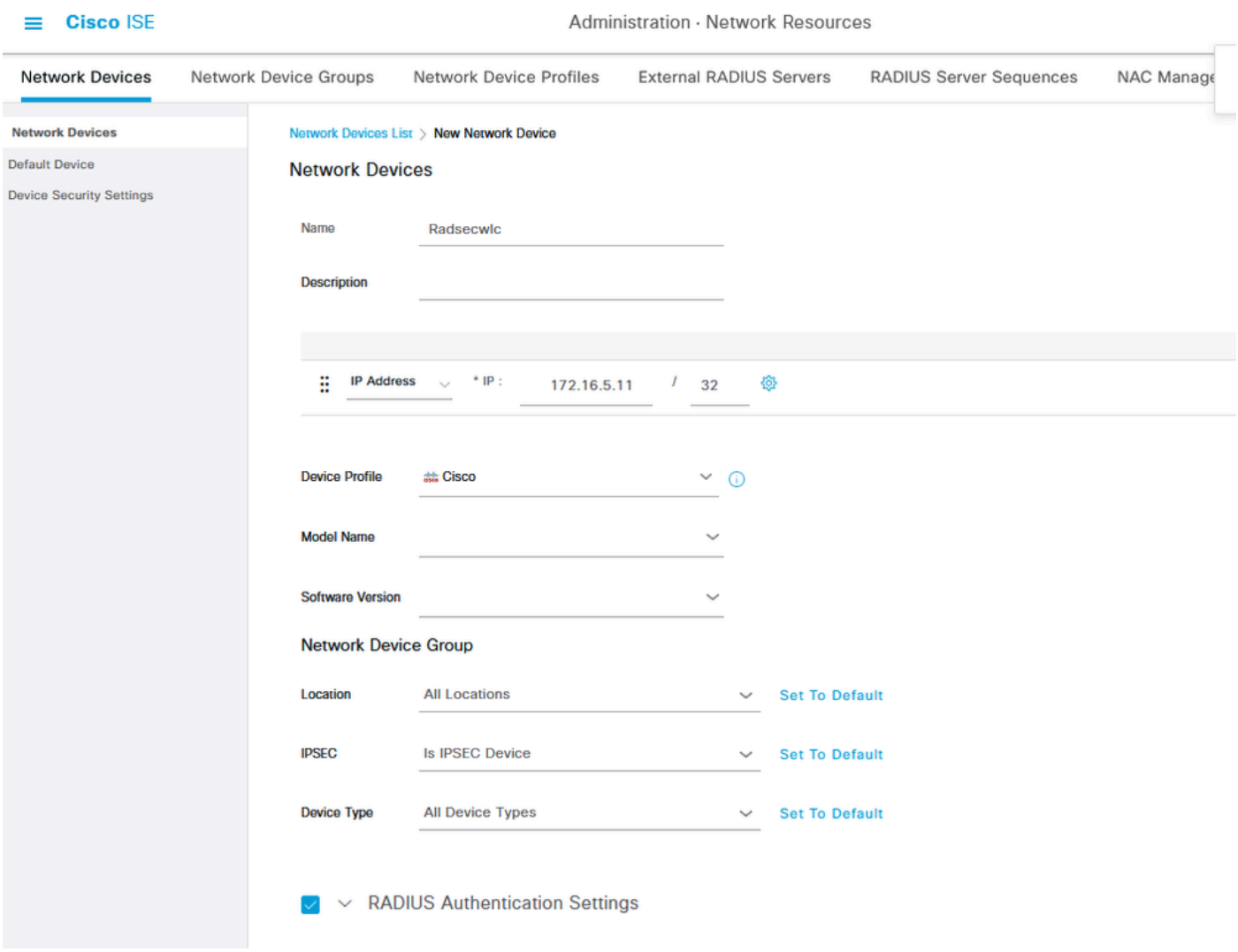
الاتصال إلى إستيراد شهادة جهاز ISE على عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) لكي تعمل RADIUS DTLS.

تكوين RADIUS DTLS

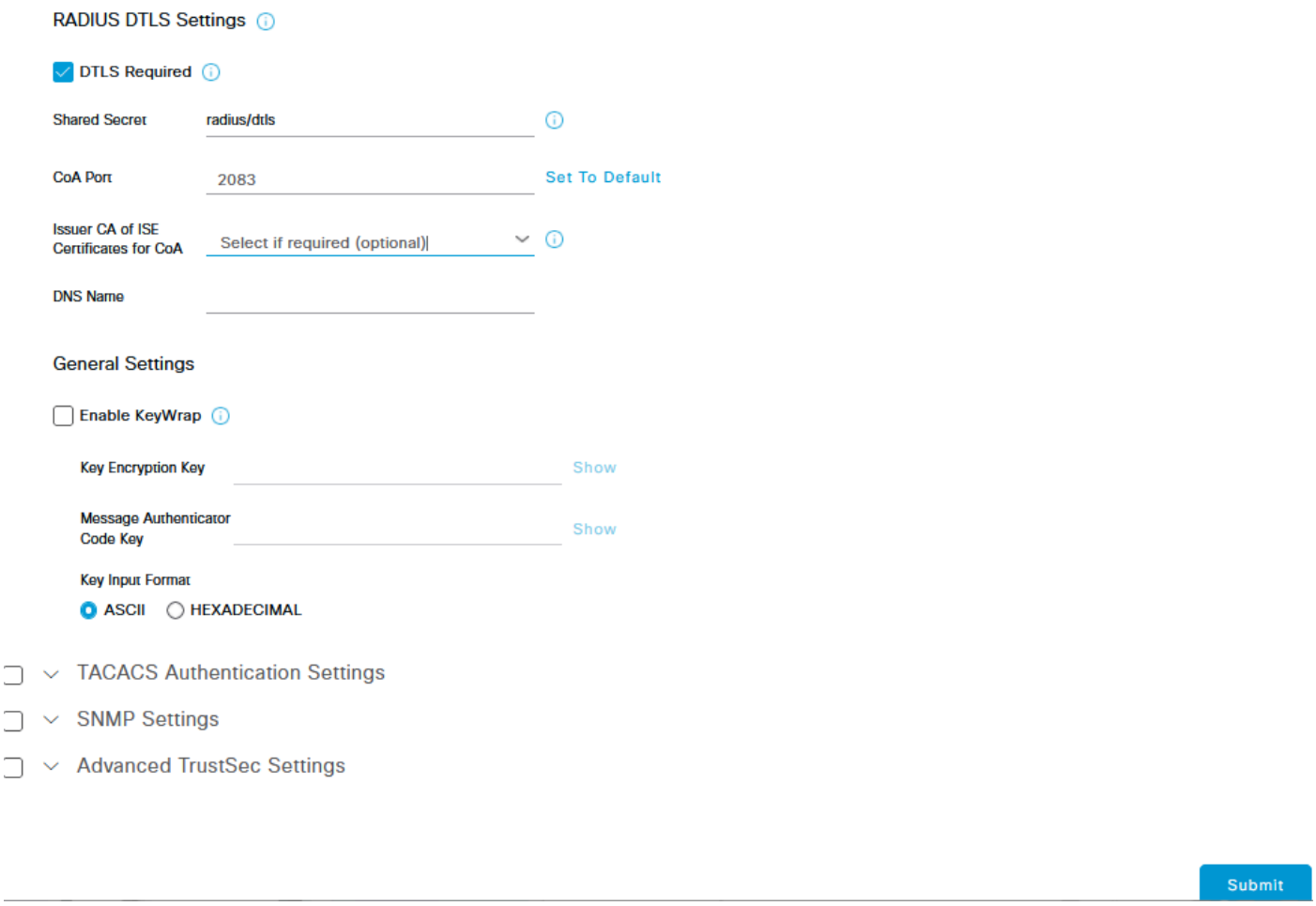
تكوين ISE

إضافة عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) كجهاز شبكة في ISE، لذلك، انتقل إلى إدارة>موارد الشبكة>أجهزة الشبكة>إضافة
أدخل اسم الجهاز وعنوان IP الخاص بواجهة WLC التي تصدر حركة مرور RADIUS. عادة ما يكون عنوان IP الخاص بواجهة الإدارة اللاسلكية. قم بالتمرير إلى أسفل والتحقق من إعدادات

مصادقة RADIUS بالإضافة إلى DTLS المطلوبة وانقر فوق إرسال:



تكوين جهاز الشبكة الجديد



إعدادات RADIUS DTLS لجهاز الشبكة على ISE

تكوين وحدة التحكّم في شبكة LAN اللاسلكية (WLC)

قم بتحديد خادم Radius جديد مع عنوان ISE IP والمنفذ الافتراضي لـ DTLS لـ RADIUS. يتوفر هذا التكوين على واجهة سطر الأوامر (CLI) فقط:

```
9800#configure terminal
9800(config)#radius server ISE
9800(config-radius-server)#address ipv4
```

```
9800(config-radius-server)#dtls port 2083
```

RADIUS DTLS ينبغي استعملت ال يشارك سر radius/dtls، ال 9800 WLC يتجاهل أي مفتاح يشكل غير هذا واحد:

```
9800(config-radius-server)#key radius/dtls
```

أستخدم الأمرdtls trustpoint client
 لتكوين TrustPoint الذي يحتوي على شهادة جهاز WLC لتبادل نفق DTLS.
أستخدم الأمرdtls trustpoint server
 لتكوين TrustPoint التي تحتوي على المرجع المصدري لشهادة جهاز ISE.

لا يكون لك من اسم العميل واسم نقطة الثقة للخادم متماثلين إلا إذا تم إصدار شهادات WLC و ISE بواسطة المرجع المصدق نفسه:

```
9800(config-radius-server)#dtls trustpoint client WLC.pfx
9800(config-radius-server)#dtls trustpoint server WLC.pfx
```

قم بتكوين عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) للتحقق من وجود أحد أسماء المواضيع البديلة (SANs) على شهادة ISE. يجب أن يتطابق هذا التكوين تماما مع أحد شبكات منطقة التخزين (SAN) الموجودة في حقل شبكات منطقة التخزين (SANs) من الشهادة.

لا ينجز ال 9800 WLC مطابقة عادية مستندة إلى التعبير لحقل SAN. هذا يعني، على سبيل المثال، أن الأمر dtls match-server-identity hostname *.example.com لشهادة حرف بدل التي تحتوي على *.example.com على حقل SAN الخاص بها صحيح ولكن الأمر نفسه للشهادة التي تحتوي على www.example.com على حقل SAN غير صحيح.

لا يتحقق عنصر التحكم في الشبكة المحلية اللاسلكية من هذا الاسم مقابل أي خادم اسم:

```
9800(config-radius-server)#dtls match-server-identity hostname ISE.example.com
9800(config-radius-server)#exit
```

إنشاء مجموعة خوادم جديدة باستخدام RADIUS DTLS الجديد للمصادقة:

```
9800(config)#aaa group server radius Radsec
9800(config-sg-radius)#server name ISE
9800(config-sg-radius)#exit
```

من هذه النقطة فصاعدا يمكنك إستخدام مجموعة الخوادم هذه كما تستخدم أي مجموعة خوادم

أخرى على عنصر التحكم في الشبكة المحلية اللاسلكية (WLC). ارجع إلى تكوين مصادقة 802.1X على سلسلة وحدة التحكم اللاسلكية Catalyst 9800 لاستخدام هذا الخادم لمصادقة العميل اللاسلكي.

التحقق من الصحة

التحقق من معلومات الشهادة

للتحقق من معلومات الشهادة للشهادات التي تم إنشاؤها، على محطة ليونكس الطرفية قم بتشغيل الأمر:

```
openssl x509 -in

      -text -noout
```

إنه يعرض معلومات الشهادة كاملة. في ذلك في تحديد المرجع المصدري لشهادة معينة وأ إذا كانت الشهادات تحتوي على وحدات الكمية القابلة للنفخ (EKU) وشبكات التخزين (SAN) المطلوبة:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = MX, ST = CDMX, L = CDMX, O = Cisco lab, OU = Cisco Wireless, CN = Intermediate.example.com
        Validity
            Not Before: Jul 18 19:14:57 2024 GMT
            Not After : Apr 14 19:14:57 2027 GMT
        Subject: C = MX, ST = CDMX, L = CDMX, O = Cisco lab, OU = Cisco Wireless, CN = WLC.example.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:b1:10:7d:6c:6c:14:2f:18:a6:0b:69:d9:60:03:
                    56:2d:48:22:f0:42:10:65:44:24:3b:54:e1:4b:87:
                    b8:ab:c5:5f:f6:a1:a3:5e:f6:3c:c5:45:cc:01:6d:
                    df:e8:a7:81:28:50:44:54:4c:af:a0:56:cf:06:be:
                    10:7e:e2:46:42:ea:3c:b9:d4:03:75:08:84:70:36:
                    bb:3d:95:3b:e2:86:e6:f7:d9:4d:00:28:c4:3c:cb:
                    f8:6d:37:5c:89:28:c1:75:b1:7e:fa:bd:91:cf:8e:
                    5c:a2:37:4f:71:da:6a:04:ee:ba:68:bf:4d:f2:d3:
                    ae:aa:13:42:3b:ff:a0:b3:65:c9:ff:f6:9a:06:d7:
                    6c:08:10:e0:b9:d8:ca:93:2d:e5:5d:7b:74:cd:93:
                    68:b1:46:c7:35:d7:6b:0f:a6:ae:34:e6:23:d1:c8:
                    d3:bf:c0:85:ab:2d:02:a8:dd:54:77:e3:32:61:4e:
                    33:58:b0:62:12:82:42:ae:2b:69:f0:5f:0c:90:c7:
                    9c:ef:b9:9c:fc:29:e2:2c:cb:b4:a9:01:fa:5d:3c:
                    97:11:67:cc:25:96:01:3d:26:1a:43:34:bd:43:b0:
                    a0:f1:ec:a0:c7:98:ad:32:32:99:9c:6b:61:af:57:
                    53:ee:20:cc:d5:ed:db:1c:5c:65:51:42:8c:28:bf:
                    62:bf
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Key Identifier:
                87:89:CA:28:06:95:D5:CE:7C:66:B4:75:81:AA:D4:19:EC:43:01:BB
            X509v3 Authority Key Identifier:
                keyid:2B:08:D8:4C:23:72:5B:62:03:EA:44:F6:9E:D9:F7:75:2E:64:97:DE
                DirName:/C=MX/ST=CDMX/L=CDMX/O=Cisco lab/OU=Cisco Wireless/CN=RootCA
                serial:01
            X509v3 Extended Key Usage:
                TLS Web Server Authentication, TLS Web Client Authentication
            X509v3 Subject Alternative Name:
                DNS:WLC.example.com, DNS:WLC2.example.com
    Signature Algorithm: sha256WithRSAEncryption
    Signature Value:
```

معلومات شهادة جهاز Cisco IOS XE كما هو موضح باستخدام OpenSSL

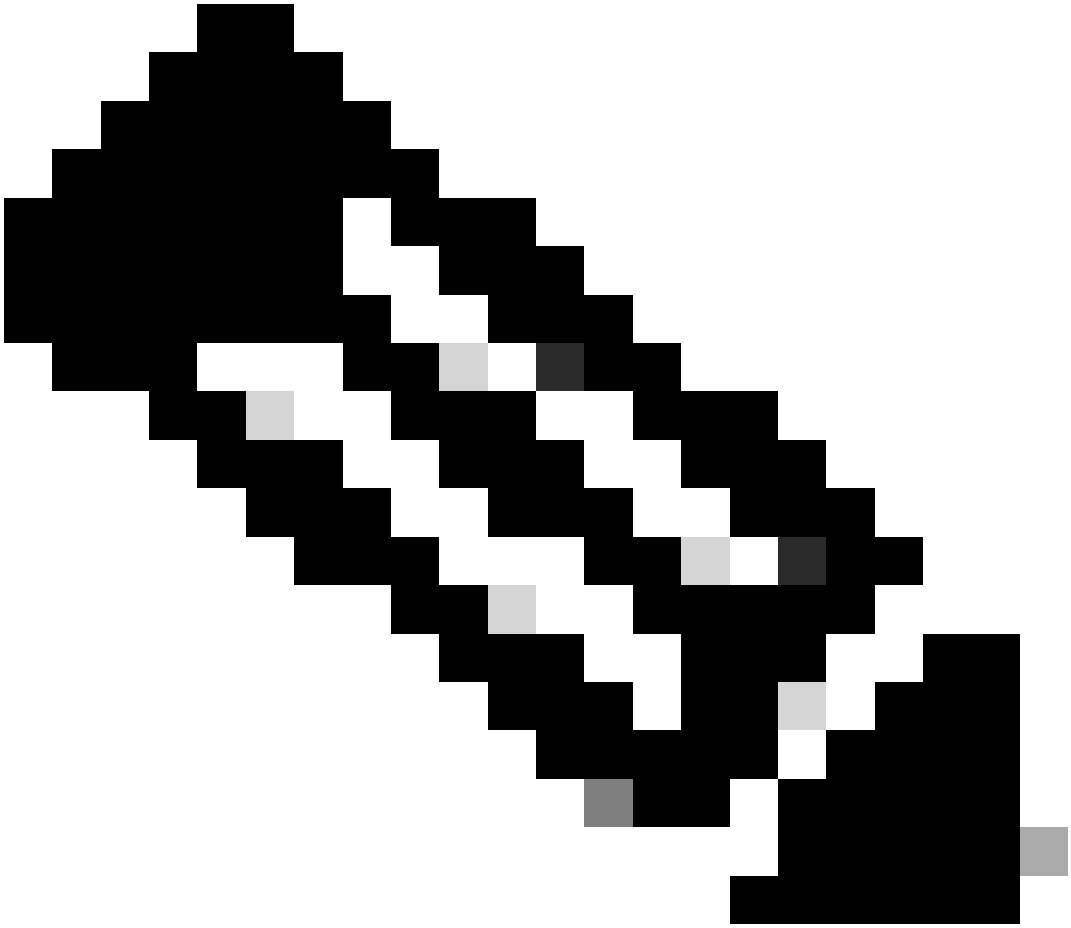
إجراء إختبار المصادقة

من عنصر التحكم في الشبكة المحلية اللاسلكية (WLC)، يمكن إختبار وظيفة DTLS لبروتوكول Radius باستخدام الأمر test aaa group

new-code

```
9800#test aaa group Radsec testuser Cisco123 new-code
User successfully authenticated
```

```
USER ATTRIBUTES

username              0   "testuser"
```



ملاحظة: ينتج رفض الوصول على أمر الاختبار يعني أن عنصر التحكم في الشبكة (WLC) يستلم رسالة RADIUS الخاصة ب Access-Reject في هذه الحالة يعمل DTLS ل RADIUS. ومع ذلك، يمكن أن تشير أيضا إلى فشل إنشاء نفق DTLS. الميزة أمر الاختبار بين السيناريوهين، راجع قسم أستكشاف الأخطاء وإصلاحها لتحديد ما إذا كانت هناك مشكلة.

استكشاف الأخطاء وإصلاحها

لمراجعة سبب فشل المصادقة يمكنك تمكين هذه الأوامر قبل إجراء إختبار المصادقة.

```
9800#debug radius
9800#debug radius radsec
9800#terminal monitor
```

هذا هو مخرج المصادقة الناجحة مع تمكين تصحيح الأخطاء:

```
9800#test aaa group Radsec testuser Cisco123 new-code
User successfully authenticated

USER ATTRIBUTES

username             0   "testuser"
9800#
Jul 18 21:24:38.301: %PARSER-5-HIDDEN: Warning!!! ' test platform-aaa group server-group Radsec user-na
Jul 18 21:24:38.313: RADIUS/ENCODE(00000000):Orig. component type = Invalid
Jul 18 21:24:38.313: RADIUS/ENCODE(00000000): dropping service type, "radius-server attribute 6 on-for-
Jul 18 21:24:38.313: RADIUS(00000000): Config NAS IP: 0.0.0.0
Jul 18 21:24:38.313: vrfid: [65535]  ipv6 tableid : [0]
Jul 18 21:24:38.313: idb is NULL
Jul 18 21:24:38.313: RADIUS(00000000): Config NAS IPv6: ::
Jul 18 21:24:38.313: RADIUS(00000000): sending
Jul 18 21:24:38.313: RADIUS/DECODE(00000000): There is no General DB. Want server details may not be sp
Jul 18 21:24:38.313: RADSEC: DTLS default secret
Jul 18 21:24:38.313: RADIUS/ENCODE: Best Local IP-Address 172.16.5.11 for Radius-Server 172.16.18.123
Jul 18 21:24:38.313: RADSEC: DTLS default secret
Jul 18 21:24:38.313: RADIUS(00000000): Send Access-Request to 172.16.18.123:2083 id 53808/10, len 54
RADIUS:  authenticator C3 4E 34 0A 91 EF 42 53 - 7E C8 BB 50 F3 98 B3 14
Jul 18 21:24:38.313: RADIUS:  User-Password       [2]   18  *
Jul 18 21:24:38.313: RADIUS:  User-Name           [1]   10  "testuser"
Jul 18 21:24:38.313: RADIUS:  NAS-IP-Address      [4]   6   172.16.5.11
Jul 18 21:24:38.313: RADIUS_RADSEC_ENQ_WAIT_Q: Success Server(172.16.18.123)/Id(10)
Jul 18 21:24:38.313: RADIUS_RADSEC_CLIENT_PROCESS: Got DATA SEND MSG
Jul 18 21:24:38.313: RADIUS_RADSEC_SOCK_SET:  0 Success
Jul 18 21:24:38.313: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 18 21:24:38.313: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 18 21:24:38.313: RADIUS_RADSEC_HASH_KEY_ADD_CTX: add [radius_radsec ctx(0x7522CE91BAC0)] succeedd f
Jul 18 21:24:38.313: RADIUS_RADSEC_GET_SOURCE_ADDR: Success
Jul 18 21:24:38.313: RADIUS_RADSEC_GET_SOCK_ADDR: Success
Jul 18 21:24:38.313: RADIUS_RADSEC_SET_LOCAL_SOCK: Success
Jul 18 21:24:38.313: RADIUS_RADSEC_SOCK_SET: Success
Jul 18 21:24:38.314: RADIUS_RADSEC_BIND_SOCKET: Success
Jul 18 21:24:38.314: RADIUS_RADSEC_CONN_SET_LPORT: Success
Jul 18 21:24:38.314: RADIUS_RADSEC_CONN_SET_SERVER_PORT: Success
Jul 18 21:24:38.314: RADIUS_RADSEC_CLIENT_HS_START: local port = 54509
Jul 18 21:24:38.314: RADIUS_RADSEC_SOCKET_CONNECT: Success
Jul 18 21:24:38.315: RADIUS_RADSEC_UPDATE_SVR_REF_CNT: Got radsec_data
Jul 18 21:24:38.315: RADIUS_RADSEC_UPDATE_SVR_REF_CNT: Got valid rctx, with server_handle B0000019
Jul 18 21:24:38.316: RADIUS_RADSEC_CLIENT_HS_START: TLS handshake in progress...(172.16.18.123/2083)
Jul 18 21:24:38.316: RADIUS_RADSEC_START_CONN_TIMER: Started (172.16.18.123/2083) for 5 secs
Jul 18 21:24:38.316: RADIUS_RADSEC_CONN_STATE_UPDATE: Success - State = 2
Jul 18 21:24:38.318: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 18 21:24:38.318: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 18 21:24:38.318: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 18 21:24:38.318: RADIUS_RADSEC_HASH_KEY_MATCH: hashkey1(0) matches hashkey2(0) TRUE
Jul 18 21:24:38.318: RADIUS_RADSEC_HASH_KEY_GET_CTX: radius radsec  sock_ctx(0x7522CE91BAC0:0) get for
Jul 18 21:24:38.318: RADIUS_RADSEC_PROCESS_SOCK_EVENT: Handle socket event for TLS handshake(172.16.18.
Jul 18 21:24:38.318: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 18 21:24:38.318: RADIUS_RADSEC_START_CONN_TIMER: Started (172.16.18.123/2083) for 5 secs
```

```
Jul 18 21:24:38.318: RADIUS_RADSEC_HS_CONTINUE: TLS handshake in progress...(172.16.18.123/2083)
Jul 18 21:24:38.318: RADIUS_RADSEC_SOCK_TLS_EVENT_HANDLE: Success
Jul 18 21:24:38.318: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 18 21:24:38.327: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 18 21:24:38.327: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 18 21:24:38.327: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 18 21:24:38.327: RADIUS_RADSEC_HASH_KEY_MATCH: hashkey1(0) matches hashkey2(0) TRUE
Jul 18 21:24:38.327: RADIUS_RADSEC_HASH_KEY_GET_CTX: radius radsec  sock_ctx(0x7522CE91BAC0:0) get for
Jul 18 21:24:38.327: RADIUS_RADSEC_PROCESS_SOCK_EVENT: Handle socket event for TLS handshake(172.16.18.
Jul 18 21:24:38.327: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 18 21:24:38.391: RADIUS_RADSEC_START_CONN_TIMER: Started (172.16.18.123/2083) for 5 secs
Jul 18 21:24:38.391: RADIUS_RADSEC_HS_CONTINUE: TLS handshake in progress...(172.16.18.123/2083)
Jul 18 21:24:38.391: RADIUS_RADSEC_SOCK_TLS_EVENT_HANDLE: Success
Jul 18 21:24:38.391: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 18 21:24:38.397: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 18 21:24:38.397: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 18 21:24:38.397: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 18 21:24:38.397: RADIUS_RADSEC_HASH_KEY_MATCH: hashkey1(0) matches hashkey2(0) TRUE
Jul 18 21:24:38.397: RADIUS_RADSEC_HASH_KEY_GET_CTX: radius radsec  sock_ctx(0x7522CE91BAC0:0) get for
Jul 18 21:24:38.397: RADIUS_RADSEC_PROCESS_SOCK_EVENT: Handle socket event for TLS handshake(172.16.18.
Jul 18 21:24:38.397: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 18 21:24:38.397: RADIUS_RADSEC_HS_CONTINUE: TLS handshake success!(172.16.18.123/2083) <-------- TL
Jul 18 21:24:38.397: RADIUS_RADSEC_CONN_STATE_UPDATE: Success - State = 3
Jul 18 21:24:38.397: RADIUS_RADSEC_UPDATE_SVR_REF_CNT: Got radsec_data
Jul 18 21:24:38.397: RADIUS_RADSEC_UPDATE_SVR_REF_CNT: Got valid rctx, with server_handle B0000019
Jul 18 21:24:38.397: RADIUS-RADSEC-HS-SUCCESS: Negotiated Cipher is ECDHE-RSA-AES256-GCM-SHA384
Jul 18 21:24:38.397: RADIUS_RADSEC_START_DATA_SEND: RADSEC HS Done, Start data send (172.16.18.123/2083
Jul 18 21:24:38.397: RADIUS_RADSEC_UNQUEUE_WAIT_Q: Success Server(172.16.18.123)/Id(10)
Jul 18 21:24:38.397: RADIUS_RADSEC_MSG_SEND: RADSEC Write SUCCESS(id=10)
Jul 18 21:24:38.397: RADIUS(00000000): Started 5 sec timeout
Jul 18 21:24:38.397: RADIUS_RADSEC_UNQUEUE_WAIT_Q: Empty Server(172.16.18.123)/Id(-1)
Jul 18 21:24:38.397: RADIUS_RADSEC_START_DATA_SEND: no more data available
Jul 18 21:24:38.397: RADIUS_RADSEC_IDLE_TIMER: Started (172.16.18.123/2083)
Jul 18 21:24:38.397: RADIUS-RADSEC-HS-SUCCESS: Success
Jul 18 21:24:38.397: RADIUS_RADSEC_SOCK_TLS_EVENT_HANDLE: Success
Jul 18 21:24:38.397: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 18 21:24:38.453: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 18 21:24:38.453: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 18 21:24:38.453: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 18 21:24:38.453: RADIUS_RADSEC_HASH_KEY_MATCH: hashkey1(0) matches hashkey2(0) TRUE
Jul 18 21:24:38.453: RADIUS_RADSEC_HASH_KEY_GET_CTX: radius radsec  sock_ctx(0x7522CE91BAC0:0) get for
Jul 18 21:24:38.453: RADIUS_RADSEC_MSG_RECV: RADSEC Bytes read= 20, Err= 0
Jul 18 21:24:38.453: RADIUS_RADSEC_SOCK_READ_EVENT_HANDLE: Radius length is 113
Jul 18 21:24:38.453: RADIUS_RADSEC_SOCK_READ_EVENT_HANDLE: Going to read rest 93 bytes
Jul 18 21:24:38.453: RADIUS_RADSEC_MSG_RECV: RADSEC Bytes read= 93, Err= 0
Jul 18 21:24:38.453: RADIUS_RADSEC_SOCK_READ_EVENT_HANDLE: linktype = 7 - src port = 2083 - dest port =
Jul 18 21:24:38.453: RADIUS: Received from id 54509/10 172.16.18.123:2083, Access-Accept, len 113 <----
RADIUS:  authenticator 4E CE 96 63 41 4B 43 04 - C7 A2 B5 05 C2 78 A7 0D
Jul 18 21:24:38.453: RADIUS:  User-Name           [1]   10  "testuser"
Jul 18 21:24:38.453: RADIUS:  Class               [25]  83
RADIUS:   43 41 43 53 3A 61 63 31 30 31 32 37 62 64 38 74  [CACS:ac10127bd8t]
RADIUS:   47 58 50 47 4E 63 6C 57 76 2F 39 67 44 66 51 67  [GXPGNclWv/9gDfQg]
RADIUS:   63 4A 76 6C 35 47 72 33 71 71 47 36 4C 66 35 59  [cJvl5Gr3qqG6Lf5Y]
RADIUS:   52 42 2F 7A 57 55 39 59 3A 69 73 65 2D 76 62 65  [RB/zWU9Y:ise-vbe]
RADIUS:   74 61 6E 63 6F 2F 35 31 30 34 33 39 38 32 36 2F  [tanco/510439826/]
RADIUS:   39                  [ 9]
Jul 18 21:24:38.453: RADSEC: DTLS default secret
Jul 18 21:24:38.453: RADIUS/DECODE(00000000): There is no General DB. Reply server details may not be r
Jul 18 21:24:38.453: RADIUS(00000000): Received from id 54509/10
```

CA غير معروف تم الإعلام عنه بواسطة WLC

عندما تستطيع عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) التحقق من الشهادات المقدمة من قبل ISE، فإنها تفشل في إنشاء نفق DTLS وتفشل عمليات المصادقة.
هذه عينة من رسائل تصحيح الأخطاء المقدمة عندما تكون هذه الحالة:

```
9800#test aaa group Radsec testuser Cisco123 new-code

Jul 19 00:59:09.695: %PARSER-5-HIDDEN: Warning!!! ' test platform-aaa group server-group Radsec user-na
Jul 19 00:59:09.706: RADIUS/ENCODE(00000000):Orig. component type = Invalid
Jul 19 00:59:09.707: RADIUS/ENCODE(00000000): dropping service type, "radius-server attribute 6 on-for-
Jul 19 00:59:09.707: RADIUS(00000000): Config NAS IP: 0.0.0.0
Jul 19 00:59:09.707: vrfid: [65535]  ipv6 tableid : [0]
Jul 19 00:59:09.707: idb is NULL
Jul 19 00:59:09.707: RADIUS(00000000): Config NAS IPv6: ::
Jul 19 00:59:09.707: RADIUS(00000000): sending
Jul 19 00:59:09.707: RADIUS/DECODE(00000000): There is no General DB. Want server details may not be sp
Jul 19 00:59:09.707: RADSEC: DTLS default secret
Jul 19 00:59:09.707: RADIUS/ENCODE: Best Local IP-Address 172.16.5.11 for Radius-Server 172.16.18.123
Jul 19 00:59:09.707: RADSEC: DTLS default secret
Jul 19 00:59:09.707: RADIUS(00000000): Send Access-Request to 172.16.18.123:2083 id 52764/13, len 54
RADIUS:  authenticator E8 09 1D B0 72 50 17 E6 - B4 27 F6 E3 18 25 16 64
Jul 19 00:59:09.707: RADIUS:  User-Password       [2]   18  *
Jul 19 00:59:09.707: RADIUS:  User-Name           [1]   10  "testuser"
Jul 19 00:59:09.707: RADIUS:  NAS-IP-Address      [4]   6   172.16.5.11
Jul 19 00:59:09.707: RADIUS_RADSEC_ENQ_WAIT_Q: Success Server(172.16.18.123)/Id(13)
Jul 19 00:59:09.707: RADIUS_RADSEC_CLIENT_PROCESS: Got DATA SEND MSG
Jul 19 00:59:09.707: RADIUS_RADSEC_SOCK_SET:  0 Success
Jul 19 00:59:09.707: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 19 00:59:09.707: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 19 00:59:09.707: RADIUS_RADSEC_HASH_KEY_ADD_CTX: add [radius_radsec ctx(0x7522CE91BAC0)] succeedd f
Jul 19 00:59:09.707: RADIUS_RADSEC_GET_SOURCE_ADDR: Success
Jul 19 00:59:09.707: RADIUS_RADSEC_GET_SOCK_ADDR: Success
Jul 19 00:59:09.707: RADIUS_RADSEC_SET_LOCAL_SOCK: Success
Jul 19 00:59:09.707: RADIUS_RADSEC_SOCK_SET: Success
Jul 19 00:59:09.707: RADIUS_RADSEC_BIND_SOCKET: Success
Jul 19 00:59:09.707: RADIUS_RADSEC_CONN_SET_LPORT: Success
Jul 19 00:59:09.707: RADIUS_RADSEC_CONN_SET_SERVER_PORT: Success
Jul 19 00:59:09.707: RADIUS_RADSEC_CLIENT_HS_START: local port = 49556
Jul 19 00:59:09.707: RADIUS_RADSEC_SOCKET_CONNECT: Success
Jul 19 00:59:09.709: RADIUS_RADSEC_UPDATE_SVR_REF_CNT: Got radsec_data
Jul 19 00:59:09.709: RADIUS_RADSEC_UPDATE_SVR_REF_CNT: Got valid rctx, with server_handle B0000019
Jul 19 00:59:09.709: RADIUS_RADSEC_CLIENT_HS_START: TLS handshake in progress...(172.16.18.123/2083)
Jul 19 00:59:09.709: RADIUS_RADSEC_START_CONN_TIMER: Started (172.16.18.123/2083) for 5 secsUser reject

uwu-9800#
Jul 19 00:59:09.709: RADIUS_RADSEC_CONN_STATE_UPDATE: Success - State = 2
Jul 19 00:59:09.711: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 19 00:59:09.711: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 19 00:59:09.711: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 19 00:59:09.711: RADIUS_RADSEC_HASH_KEY_MATCH: hashkey1(0) matches hashkey2(0) TRUE
Jul 19 00:59:09.711: RADIUS_RADSEC_HASH_KEY_GET_CTX: radius radsec  sock_ctx(0x7522CE91BAC0:0) get for
Jul 19 00:59:09.711: RADIUS_RADSEC_PROCESS_SOCK_EVENT: Handle socket event for TLS handshake(172.16.18.
Jul 19 00:59:09.711: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 19 00:59:09.711: RADIUS_RADSEC_START_CONN_TIMER: Started (172.16.18.123/2083) for 5 secs
Jul 19 00:59:09.711: RADIUS_RADSEC_HS_CONTINUE: TLS handshake in progress...(172.16.18.123/2083)
Jul 19 00:59:09.711: RADIUS_RADSEC_SOCK_TLS_EVENT_HANDLE: Success
```

```
Jul 19 00:59:09.713: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 19 00:59:09.720: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 19 00:59:09.720: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 19 00:59:09.720: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 19 00:59:09.720: RADIUS_RADSEC_HASH_KEY_MATCH: hashkey1(0) matches hashkey2(0) TRUE
Jul 19 00:59:09.720: RADIUS_RADSEC_HASH_KEY_GET_CTX: radius radsec  sock_ctx(0x7522CE91BAC0:0) get for
Jul 19 00:59:09.720: RADIUS_RADSEC_PROCESS_SOCK_EVENT: Handle socket event for TLS handshake(172.16.18.
Jul 19 00:59:09.720: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 19 00:59:09.722: RADIUS_RADSEC_HS_CONTINUE: TLS handshake failed!
Jul 19 00:59:09.722: RADIUS_RADSEC_UNQUEUE_WAIT_Q: Success Server(172.16.18.123)/Id(13)
Jul 19 00:59:09.722: RADIUS_RADSEC_FAILOVER_HANDLER:Failng-over to new server = 0x0
Jul 19 00:59:09.722: RADIUS_RADSEC_UNQUEUE_WAIT_Q: Empty Server(172.16.18.123)/Id(-1)
Jul 19 00:59:09.722: RADIUS_RADSEC_FAILOVER_HANDLER: no more data available
Jul 19 00:59:09.722: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 19 00:59:09.722: RADIUS_RADSEC_CONN_CLOSE: Cleaned up timers for Radius RADSEC ctx
Jul 19 00:59:09.722: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 19 00:59:09.722: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 19 00:59:09.722: RADIUS_RADSEC_HASH_KEY_DEL_CTX: remove [radius_radsec  ctx(0x7522CE91BAC0)] succee
Jul 19 00:59:09.722: RADIUS_RADSEC_CONN_CLOSE: Hash table entry removed for RADSEC sock ctx
Jul 19 00:59:09.723: RADIUS_RADSEC_CONN_CLOSE: Success
Jul 19 00:59:09.723: RADIUS_RADSEC_SOCK_TLS_EVENT_HANDLE: Failed to complete TLS handshake <----------D
Jul 19 00:59:09.723: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 19 00:59:09.723: RADIUS_RADSEC_CONN_CLOSE: Cleaned up timers for Radius RADSEC ctx
Jul 19 00:59:09.723: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(-1) generated for sock(-1)
Jul 19 00:59:09.723: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(-1) generated for sock(-1)
uwu-9800#
Jul 19 00:59:09.723: RADIUS_RADSEC_HASH_KEY_DEL_CTX: remove [radius_radsec  ctx(0x7522CE91BAC0)] succee
Jul 19 00:59:09.723: RADIUS_RADSEC_CONN_CLOSE: Hash table entry removed for RADSEC sock ctx
Jul 19 00:59:09.723: RADIUS_RADSEC_CONN_CLOSE: Success
Jul 19 00:59:09.723: RADIUS_RADSEC_SOCK_TLS_EVENT_HANDLE: Error
Jul 19 00:59:09.723: RADIUS_RADSEC_PROCESS_SOCK_EVENT: failed to hanlde radsec hs event
Jul 19 00:59:09.723: RADIUS/DECODE: No response from radius-server; parse response; FAIL
Jul 19 00:59:09.723: RADIUS/DECODE: Case error(no response/ bad packet/ op decode);parse response; FAIL
Jul 19 00:59:09.723: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
Jul 19 00:59:10.718: %RADSEC_AUDIT_MESSAGE-3-FIPS_AUDIT_FCS_RADSEC_SERVER_CERTIFICATE_VALIDATION_FAILUR
Jul 19 00:59:10.718: %RADSEC_AUDIT_MESSAGE-3-FIPS_AUDIT_FCS_RADSEC_SERVER_IDENTITY_CHECK_FAILURE: Chass
Jul 19 00:59:10.718: %RADSEC_AUDIT_MESSAGE-6-FIPS_AUDIT_FCS_DTLSC_DTLS_SESSION_CLOSED: Chassis 1 R0/0:
```

لتصحيح ذلك، تأكد من أن الهوية التي تم تكوينها على عنصر التحكم في الشبكة (WLC) المحلية اللاسلكية تطابق تمامًا إحدى شهادات التخزين (SAN) المضمنة في شهادة ISE:

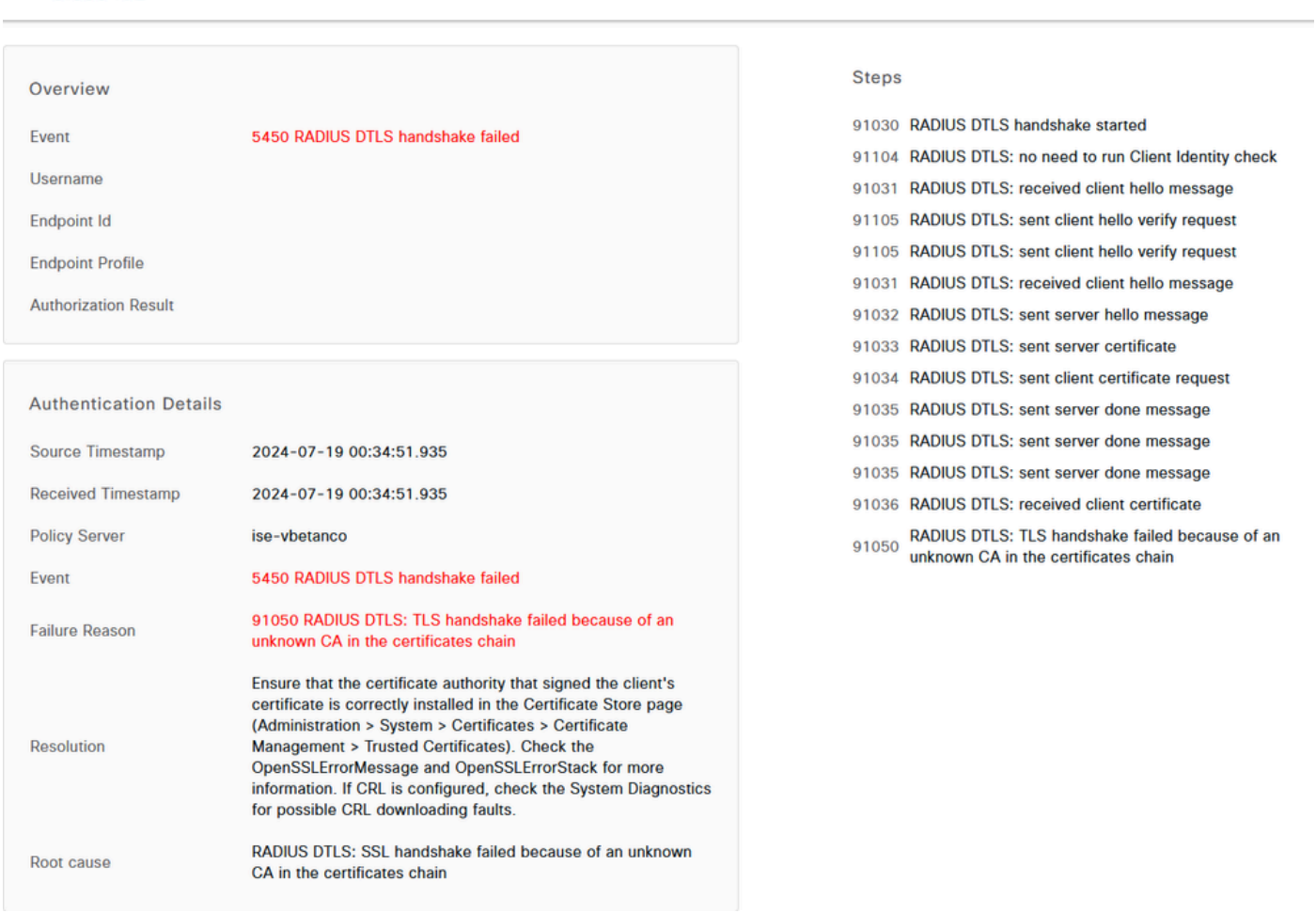
```
9800(config)#radius server
```

```
9800(config)#dtls match-server-identity hostname
```

تأكد من إستيراد سلسلة شهادات CA بشكل صحيح على وحدة التحكم وأن **dtls trustpoint server** configuration uses the Issuer CA trustpoint.

CA غير معروف تم الإعلام عنه بواسطة ISE

عندما يتعذر على ISE التحقق من صحة الشهادات المقدمة من قبل عنصر التحكم في الشبكة المحلية اللاسلكية (WLC)، فإنه يفشل في إنشاء نفق DTLS وتفشل المصادقات. يظهر هذا كخطأ في سجلات RADIUS المباشرة. انتقل إلى Operations>سجلات Radius>المباشرة للتحقق.



يبلغ سجل ISE Live عن فشل مصافحة DTLS بسبب CA غير معروف

لتصحيحه، تأكد من كل من الشهادات الوسيطة والجذرية، حدد خانات الاختيار ثقة مصادقة العميل و syslog تحت إدارة>نظام>الشهادات>الشهادات الموثوق بها.

التحقق من الإبطال موجود في المكان

عندما يتم إستيراد الشهادات إلى عنصر التحكم في الشبكة المحلية اللاسلكية (WLC)، فإن نقاط الثقة التي تم إنشاؤها حديثا تحتوي على فحص إبطال ممكن. يؤدي ذلك إلى محاولة عنصر التحكم في الشبكة المحلية اللاسلكية (WLC) البحث عن قائمة إبطال الشهادات غير المتوفرة أو التي يمكن الوصول إليها ويفشل التحقق من الشهادة.
تأكد من أن كل نقطة ثقة في مسار التحقق من الشهادات تحتوي على الأمر revocation-check none .

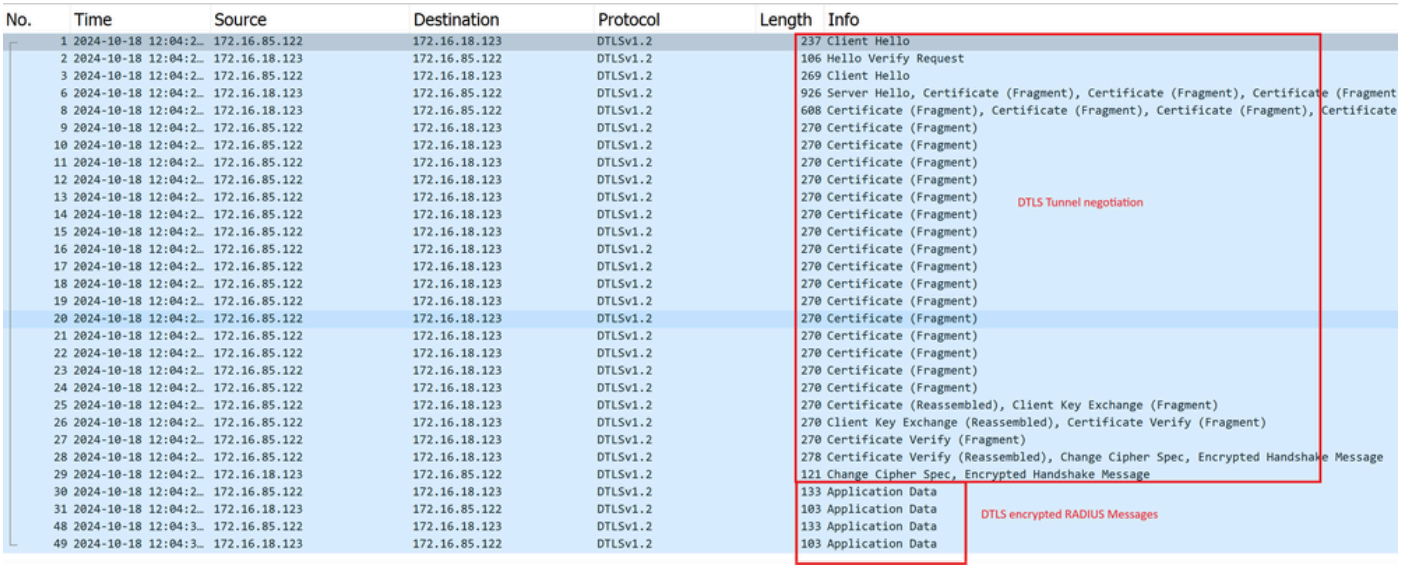
```
Jul 17 21:50:39.064: RADIUS_RADSEC_HASH_KEY_MATCH: hashkey1(0) matches hashkey2(0) TRUE
Jul 17 21:50:39.064: RADIUS_RADSEC_HASH_KEY_GET_CTX: radius radsec  sock_ctx(0x780FB0715978:0) get for
Jul 17 21:50:39.064: RADIUS_RADSEC_PROCESS_SOCK_EVENT: Handle socket event for TLS handshake(172.16.18.
Jul 17 21:50:39.064: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 17 21:50:39.068: %PKI-3-CRL_FETCH_FAIL: CRL fetch for trustpoint WLC1.pfx failed
                      Reason : Enrollment URL not configured. <------ WLC tries to perform revocation ch
Jul 17 21:50:39.070: RADIUS_RADSEC_HS_CONTINUE: TLS handshake failed!
Jul 17 21:50:39.070: RADIUS_RADSEC_UNQUEUE_WAIT_Q: Success Server(172.16.18.123)/Id(2)
Jul 17 21:50:39.070: RADIUS_RADSEC_FAILOVER_HANDLER:Failng-over to new server = 0x0
Jul 17 21:50:39.070: RADIUS_RADSEC_UNQUEUE_WAIT_Q: Empty Server(172.16.18.123)/Id(-1)
Jul 17 21:50:39.070: RADIUS_RADSEC_FAILOVER_HANDLER: no more data available
Jul 17 21:50:39.070: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 17 21:50:39.070: RADIUS_RADSEC_CONN_CLOSE: Cleaned up timers for Radius RADSEC ctx
Jul 17 21:50:39.070: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(0) generated for sock(0)
Jul 17 21:50:39.070: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(0) generated for sock(0)
Jul 17 21:50:39.070: RADIUS_RADSEC_HASH_KEY_DEL_CTX: remove [radius_radsec  ctx(0x780FB0715978)] succee
Jul 17 21:50:39.070: RADIUS_RADSEC_CONN_CLOSE: Hash table entry removed for RADSEC sock ctx
Jul 17 21:50:39.070: RADIUS_RADSEC_CONN_CLOSE: Success
Jul 17 21:50:39.070: RADIUS_RADSEC_SOCK_TLS_EVENT_HANDLE: Failed to complete TLS handshake
Jul 17 21:50:39.070: RADIUS_RADSEC_STOP_TIMER: Stopped (172.16.18.123/2083)
Jul 17 21:50:39.070: RADIUS_RADSEC_CONN_CLOSE: Cleaned up timers for Radius RADSEC ctx
Jul 17 21:50:39.070: RADIUS_RADSEC_GENERATE_HASHKEY: hash key(-1) generated for sock(-1)
Jul 17 21:50:39.070: RADIUS_RADSEC_GENERATE_HASHBUCKET: hash bucket(-1) generated for sock(-1)
Jul 17 21:50:39.070: RADIUS_RADSEC_HASH_KEY_DEL_CTX: remove [radius_radsec  ctx(0x780FB0715978)] succee
Jul 17 21:50:39.070: RADIUS_RADSEC_CONN_CLOSE: Hash table entry removed for RADSEC sock ctx
Jul 17 21:50:39.070: RADIUS_RADSEC_CONN_CLOSE: Success
Jul 17 21:50:39.070: RADIUS_RADSEC_SOCK_TLS_EVENT_HANDLE: Error
Jul 17 21:50:39.070: RADIUS_RADSEC_PROCESS_SOCK_EVENT: failed to hanlde radsec hs event
Jul 17 21:50:39.070: RADIUS_RADSEC_CLIENT_PROCESS: Got Socket Event
```

أستكشاف أخطاء إنشاء نفق DTLS وإصلاحها على التقاط الحزمة

توفر وحدة التحكم في الشبكة المحلية اللاسلكية (WLC) طراز 9800 ميزة التقاط الحزم المضمنة (EPC) التي تتيح لك التقاط حركة مرور البيانات التي تم إرسالها واستقبالها واصدارة. عند إستخدامها في نفس الوقت، فإنها تسمح لك بتحليل حركة مرور إنشاء جلسة DTLS من منظور الك الجهازين.

يرجى الرجوع إلى دليل مسؤول محرك خدمات الهوية من Cisco للحصول على خطوات تفصيلية لتكوين تفريغ TCP على ISE. راجع أيضا وحدات التحكم في الشبكة المحلية (LAN) اللاسلكية Catalyst 9800 لتكوين معلومات حول استكشاف الأخطاء وإصلاحها للحصول على وحدة التحكم في الشبكة المحلية اللاسلكية (WLC) لميزة EPC.

هذا مثال على إنشاء نفق DTLS ناجح.



التقاط حزمة لمفاوضات نفق RADIUS DTLS والرسائل المشفرة

تظهر عمليات التقاط الحزم كيفية إنشاء نفق DTLS. إذا كانت هناك مشكلة مع التفاوض، من حركة مرور مفقودة بين الأجهزة أو حزم تنبيه DTLS المشفرة، يساعدك التقاط الحزمة على تحديد المشكلة.

حول هذه الترجمة
ترجمت Cisco هذا المستند باستخدام مجموعة من التقنيات الآلية والبشرية لتقديم محتوى دعم للمستخدمين في جميع أنحاء العالم بلغتهم الخاصة. يُرجى ملاحظة أن أفضل ترجمة آلية لن تكون دقيقة كما هو الحال مع الترجمة الاحترافية التي يقدمها مترجم محترف. تخلي Cisco Systems مسئوليتها عن دقة هذه الترجمات وتُوصي بالرجوع دائمًا إلى المستند الإنجليزي الأصلي (الرابط متوفر).