サードパーティ証明書用 CSR の生成とチェーン証明書の WLC へのダウンロード

内容

はじめに

このドキュメントでは、AireOS WLCで証明書を生成およびインポートする方法について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- 基本動作用に WLC、Lightweight アクセス ポイント（LAP）、およびワイヤレス クライアント カードを設定する方法.
- OpenSSL アプリケーションを使用する方法.
- 公開キー インフラストラクチャとデジタル証明書

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- ファームウェア バージョン 8.3.102 が稼働している Cisco 5508 WLC
- Microsoft Windows 用の OpenSSL アプリケーション
- サードパーティ認証局（CA ）に固有の登録ツール

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

チェーン証明書

証明書チェーンは、チェーン内の各証明書が後続の証明書によって署名される一連の証明書です。

証明書チェーンの目的は、ピア証明書から信頼される CA 証明書までの連鎖された信頼を築くことです。CAは、署名時にピア証明書内のIDを保証します。

その CA が信頼する CA である場合（ルート証明書ディレクトリに CA 証明書のコピーが存在することによって示されます）、署名されたピア証明書も信頼できることを意味します。

クライアントは、既知の CA によって作成された証明書でなければ受け入れないことがあります。通常、クライアントでは、証明書の妥当性を確認できないと示します。

これは、証明書の署名がクライアントのブラウザに設定されていない 中間 CA による場合です。その場合は、チェーン SSL 証明書または証明書グループを使用する必要があります。

チェーン証明書のサポート

コントローラを使用して、デバイス証明書を Web 認証用のチェーン証明書としてダウンロードできます。

証明書のレベル

- レベル 0：WLC 上のサーバ証明書のみを使用
- レベル 1：WLC 上のサーバ証明書と CA ルート証明書を使用
- レベル 2：WLC 上のサーバ証明書、1 つの署名付き中間 CA 証明書、および CA ルート証明書を使用
- レベル 3：WLC 上のサーバ証明書、2 つの署名付き中間 CA 証明書、および CA ルート証明書を使用

WLC では、サイズが 10KB を超えるチェーン証明書はサポートされません。ただし、WLC バージョン 7.0.230.0 以降では、この制限が取り除かれています。

注：チェーン証明書がサポートされており、Web認証とWeb管理に実際に必要です

注：ワイルドカード証明書は、ローカルEAP、管理、またはWeb認証で完全にサポートされています

次の任意の Web 認証証明書を使用できます。

- チェーン証明書
- チェーンされていない証明書
- 自動生成

注:WLCバージョン7.6以降では、チェーン証明書のみがサポートされています（したがって、必要です）

管理目的でチェーンされていない証明書を生成するには、このドキュメントを参照し、証明書がCA証明書と組み合わされる部分は無視してください。

このドキュメントでは、チェーン Secure Socket Layer（SSL）証明書を WLC に適切にインストールする方法を説明します。

ステップ 1：CSR の生成

CSR を生成する方法は 2 つあります。OpenSSL（8.3よりも前のWLCソフトウェアで可能な唯一の方法）を使用して手動で行うか、WLC自体でCSRを生成します（8.3.102以降で使用可能）。

オプション A：OpenSSL を使用した CSR

注:Chromeバージョン58以降は、証明書の共通名だけを信頼せず、サブジェクト代替名(SAN)も存在する必要があります。次のセクションでは、このブラウザの新しい要件であるOpenSSL CSRにSANフィールドを追加する方法について説明します。

OpenSSL を使用して CSR を生成するには、次の手順を実行します。

1. OpenSSL をインストールして開きます。

   Microsoft Windowsでは、デフォルトでopenssl.exeは次の場所にあります **C:\ > openssl > bin**を参照。

   注：古いWLCリリースにはOpenSSLバージョン0.9.8が推奨されていますが、バージョン7.5ではOpenSSLバージョン1.0のサポートも追加されており(Cisco Bug ID CSCti65315 - OpenSSL v1.0で生成された証明書のサポートが必要なことを参照)、このバージョンの使用が推奨されています。OpenSSL 1.1の動作もテスト済みで、8.x以降のWLCリリースで動作します。

2. OpenSSL の設定ファイルを見つけてコピーし、この CSR 用に編集します。コピーを編集して次のセクションを追加します（図1を参照）。

3.
   <#root>

```
[req]

req_extensions = v3_req


[ v3_req ]

# Extensions to add to a certificate request

basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment

subjectAltName = @alt_names


[alt_names]

DNS.1 = server1.example.com
DNS.2 = mail.example.com
DNS.3 = www.example.com
DNS.4 = www.sub.example.com
DNS.5 = mx.example.com
DNS.6 = support.example.com
```

   「DNS.1」、「DNS.2」（など）で始まる行には、証明書の代替名がすべて含まれている必要があります。次に、WLCで使用されるURLを書き込みます。前の例の太字の行は、ラボのopenSSLバージョンでは存在しないか、コメントになっています。オペレーティングシステムやopensslのバージョンによって大きく異なる場合があります。この変更されたバージョンの設定を次のように保存します。 **openssl-san.cnf** 使用していますが、
4. 新しいCSRを生成するには、次のコマンドを入力します。

   <#root>

```
OpenSSL>

req -new -newkey rsa:3072 -nodes -keyout mykey.pem -out myreq.pem -config openssl-san.cnf
```

   > 注:WLCでは、8.5ソフトウェアバージョンで最大キーサイズ4096ビットがサポートされています

5. 国名、州、都市などの情報の入力を求めるプロンプトが表示されます。必要な情報を入力します。

   > 注：正しい共通名を指定することが重要です。証明書の作成に使用されるホスト名（共通名）が、WLC 上の仮想インターフェイス IP アドレスのドメイン ネーム システム（DNS）のホスト名エントリに一致すること、そしてその名前が DNS にも存在す

ることを確認します。また、仮想 IP (VIP) インターフェイスへの変更後には、この変更を反映するためにシステムをリブートする必要があります。

ランダム データの例は次のとおりです。

<#root>

```
OpenSSL>

req -new -newkey rsa:3072 -nodes -keyout mykey.pem -out myreq.pem -config openssl-san.cnf


Loading 'screen' into random state - done
Generate a 1024 bit RSA private key
................................................................++++++
...............................................++++++
writing new private key to 'mykey.pem'
-----
You are about to be asked to enter information that is 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 is a default value,
If you enter '.', the field is left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:CA
Locality Name (eg, city) []:San Jose
Organization Name (eg, company) [Internet Widgits Pty Ltd]:ABC
Organizational Unit Name (eg, section) []:CDE
Common Name (eg, YOUR name) []:XYZ.ABC
Email Address []:(email address)

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:Test123
An optional company name []:OpenSSL>
```

6. CSR (特にSAN属性presencE) は、 **openssl req -text -noout -in csrfilename**
7. すべての必要な詳細を入力すると、2 つのファイルが生成されます。
   - mykey.pem という名前を含む新しい秘密キー
   - myreq.pem という名前を含む CSR

オプション B : WLC による CSR の生成

WLCがソフトウェアバージョン8.3.102以降を実行している場合、より安全なオプションは、WLCを使用してCSRを生成することです。利点は、キーがWLC上で生成され、WLCから送信されないことです。したがって、外部に公開されることはありません。

現時点では、この方法では、SAN属性の存在を必要とする特定のブラウザの問題につながることが知られているCSRにSANを設定することはできません。一部のCAでは、署名時にSANフィー

ルドを挿入できるため、CAに確認することをお勧めします。

WLC自体によるCSR生成では2048ビットのキーサイズが使用され、ecdsaキーサイズは256ビットです。

注:csr generationコマンドを実行し、それに続く証明書をまだインストールしていない場合、WLCはリブート後に新しく生成されたCSRキーを使用しますが、それに伴う証明書を持たないため、次のリブート時にWLCはHTTPSで完全に到達不能になります。

Web 認証用の CSR を生成するには、次のコマンドを入力します。

(WLC) >**config certificate generate csr-webauth BE BR Brussels Cisco TAC mywebauthportal.wireless.com tac@cisco.com**

-----BEGIN CERTIFICATE REQUEST-----
MIICqjCCAZICAQAwZTELMAkGA1UECAwCQlIxETAPBgNVBAcMCEJydXNzZWxzMQ4w
DAYDVQQKDAVDaXNjbzEMMAoGA1UECwwDVEFDMSUwIwYDVQQDDBxteXdlYmF1dGhw
b3J0YWwud2lyZWxlc3MuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKC
AQEAnssc0BxlJ2ULa3xgJH5lAUtbd9CuQVqqf2nflh+V1tu82rzTvz38bjF3g+MX
JiaBbKMA27VJH1J2K2ycDMlhjyYpH9N59T4fXvZr3JNGVfmHIRuYDnCSdil0ookK
FU4sDwXyOxR6gfB6m+Uv5SCOuzfBsTz5bfQ1NlZqg1hNemnhqVgbXEd90sgJmaF2
0tsL0jUhbLosdwMLUbZ5LUa34mvufoI3VAKA0cmWZh2WzMJiaI2JpbO0afRO3kSg
x3XDkZiR7Z9a8rK6Xd8rwDlx0TcMFWdWVcKMDgh7Tw+Ba1cUjjIMzKT6OOjFGOGu
yNkgYefrrBN+WkDdc6c55bxErwIDAQABoAAwDQYJKoZIhvcNAQELBQADggEBAB0K
ZvEpAafoovphlcXIElL2DSwVzjlbd9u7T5JRGgqri1l9/0wzxFjTymQofga427mj
5dNqlCWxRFmKhAmO0fGQkUoP1YhJRxidu+0T8O46s/stbhj9nuInmoTgPaA0s3YH
tDdWgjmV2ASnroUV9oBNu3wR6RQtKDX/CnTSRG5YufTWOVf9IRnL9LkU6pzA69Xd
YHPLnD2ygR1Q+3Is4+5Jw6ZQAaqlPWyVQccvGyFacscA7L+nZK3SSITzGt9B2HAa
PQ8DQOaCwnqt2efYmaezGiHOR8XHOaWcNoJQCFOnb4KK6/1aF/7eOS4LMA+jSzt4
Wkc/wH4DyYdH7x5jzHc=
-----END CERTIFICATE REQUEST-----

webadminのCSRを生成するために、コマンドは次のように変更されます。

(WLC) >config certificate generate csr-webadmin BE BR Brussels Cisco TAC mywebauthportal.wireless.com tac@cisco.com

---

注：コマンドを入力すると、CSRが端末に出力されます。他の方法で取得することはできません。WLCからアップロードすることも、保存することもできません。 コマンドを入力した後、コンピュータ上のファイルに CSR を コピーアンドペーストする必要があります。生成されたキーは、次の CSR が生成されるまで WLC に残ります（そのためキーは上書きされます）。後でWLCハードウェアを変更(RMA)する必要がある場合は、新しいキーと同じ証明書を再インストールできません。新しいWLCでCSRが生成されます。

から

生成された CSR は、サードパーティの署名機関またはエンタープライズ公開キーインフラスト

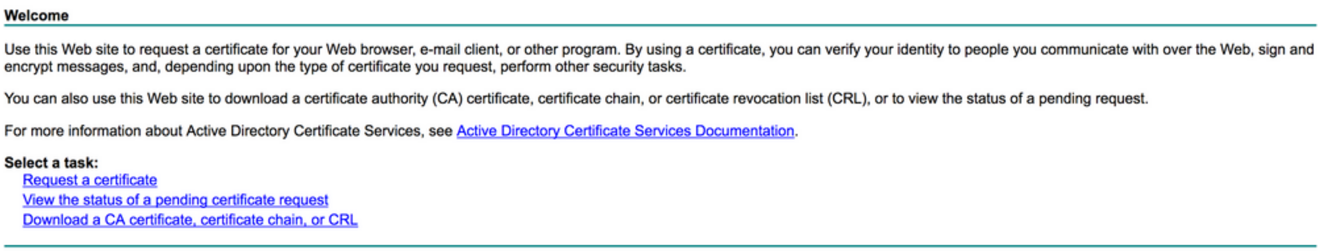
ラクチャ（PKI）に渡す必要があります。

ステップ２：署名された証明書の取得

オプションA：エンタープライズCAからFinal.pemファイルを取得する

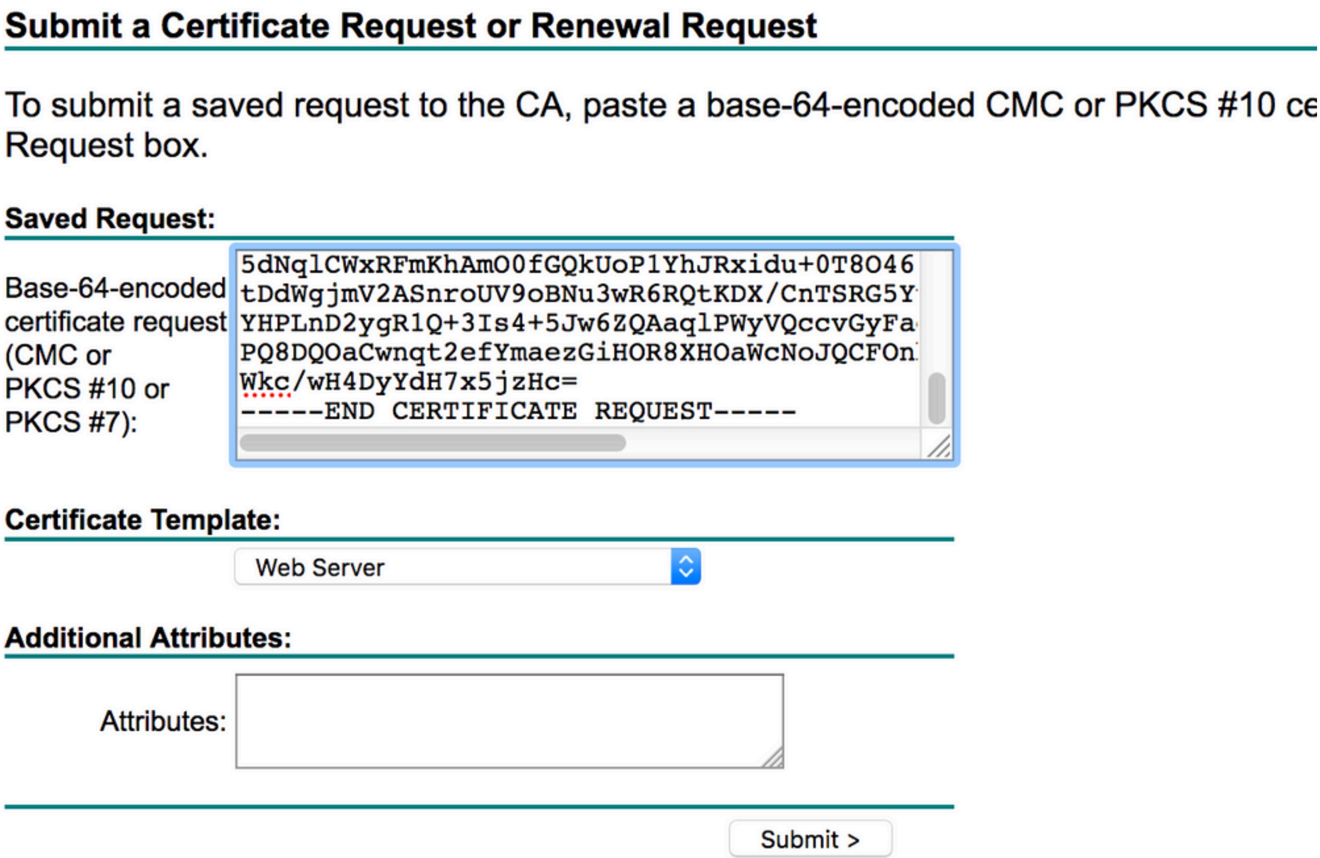
この例では、現在のエンタープライズCA（この例ではWindows Server 2012）のみを示し、Windows Server CAを最初からセットアップする手順は示しません。

1. ブラウザでエンタープライズCAページ(通常はhttps://<CA-ip>/certsrv)に移動し、 **Request a certificate**を参照。

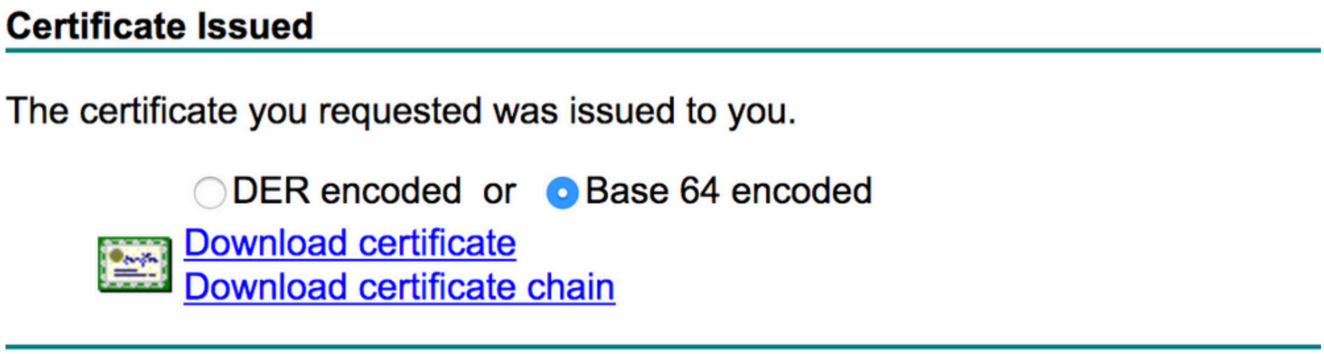


2. クリック **advanced certificate request**を参照。

3. WLC または OpenSSL から取得した CSR を入力します。Certificate Templateドロップダウンリストで、 **Web Server**を参照。



4. ポリシーの横の [レポート（Report）] **Base 64 encoded**オプションボタンを選択します。



5. ダウンロードした証明書のタイプがPKCS7(.p7b)の場合は、証明書をPEMに変換します（次の例では、証明書チェーンはファイル名「All-certs.p7b」としてダウンロードされています）。

```
openssl pkcs7 -print_certs -in All-certs.p7b -out All-certs.pem
```

6.オプションA（CSRを生成するOpenSSL）を選択した場合は、証明書チェーン（この例では「All-certs.pem」という名前）証明書とCSRとともに生成された秘密キー（デバイス証明書の秘密キー。この例ではmykey.pem）を組み合わせて、ファイルをfinal.pemとして保存します。CSRを

WLC（オプションB）から直接生成した場合は、この手順をスキップします。

All-certs.pemファイルとfinal.pemファイルを作成するには、OpenSSLアプリケーションで次のコマンドを入力します。

<#root>

```
openssl>

pkcs12 -export -in All-certs.pem -inkey mykey.pem
       -out All-certs.p12 -clcerts -passin pass:check123
       -passout pass:check123
```

```
openssl>

pkcs12 -in All-certs.p12 -out final.pem
       -passin pass:check123 -passout pass:check123
```

注：このコマンドでは、パラメータ -passin および -passoutに対してパスワードを入力する必要があります。-passout パラメータに設定するパスワードは、WLC 上で設定する certpassword パラメータと一致している必要があります。上記の例では、-passin と -passout の両方のパラメータに対して設定されるパスワードは check123 です。

Final.pemは、「Option A. CSR with OpenSSL」に従ってWLCにダウンロードするファイルです。

「オプションB. WLC自体によって生成されたCSR」に従った場合、All-certs.pemはWLCにダウンロードするファイルです。次の手順では、このファイルを WLC にダウンロードします。

注:WLCへの証明書のアップロードが失敗した場合は、PEMファイルにチェーン全体があることを確認します。これがどのように表示されるかについては、オプションBのステップ2（サードパーティCAからfinal.pemを取得する）を参照してください。ファイルで表示される証明書が 1 つだけの場合は、すべての中間 CA 証明書ファイルとルート CA 証明書ファイルを手動でダウンロードし、ファイルに追加して（単純にコピーアンドペーストして）チェーンを作成します。

オプションB：サードパーティCAからFinal.pemファイルを取得する

1. CSR の情報をコピーして、任意の CA の登録ツールに貼り付けます。

サードパーティ CA に CSR を送信すると、サードパーティ CA がデジタル署名した証明書チェーンが電子メールで返されてきます。チェーン証明書の場合、CA から証明書のチェーン全体が返されます。この例のように 中間証明書が 1 つだけ表示される場合は、CA から次の 3 種類の証明書を受け取ります。

- Root certificate.pem
- Intermediate certificate.pem
- Device certificate.pem

注：証明書がApacheとSecure Hash Algorithm 1(SHA1)暗号化に対応していることを確認してください。

2. 3 つの証明書をすべて入手したら、各 .pem ファイルの内容を以下の順で別のファイルにコピー アンド ペーストします。

```
------BEGIN CERTIFICATE------
*Device cert*
------END CERTIFICATE------
------BEGIN CERTIFICATE------
*Intermediate CA cert *
------END CERTIFICATE--------
------BEGIN CERTIFICATE------
*Root CA cert *
------END CERTIFICATE------
```

3. All-certs.pem としてファイルを保存します。

4. オプションA（CSRを生成するためにOpenSSL）を使用した場合は、All-certs.pem証明書とCSRとともに生成された秘密キー（デバイス証明書の秘密キー。この例では mykey.pem）を組み合わせて、ファイルをfinal.pemとして保存します。CSRをWLC（オプションB）から直接生成した場合は、この手順をスキップします。

   All-certs.pemファイルとfinal.pemファイルを作成するには、OpenSSLアプリケーションで次のコマンドを入力します。

<#root>

```
openssl>

pkcs12 -export -in All-certs.pem -inkey mykey.pem
       -out All-certs.p12 -clcerts -passin pass:check123
       -passout pass:check123



openssl>

pkcs12 -in All-certs.p12 -out final.pem
       -passin pass:check123 -passout pass:check123
```

注：このコマンドでは、パラメータ -passin および -passoutに対してパスワードを入力する必要があります。-passout パラメータに設定するパスワードは、WLC 上で設定する certpassword パラメータと一致している必要があります。上記の例では、-passin と -passout の両方のパラメータに対して設定されるパスワードは check123 です。

Final.pemは、「Option A. CSR with OpenSSL」に従ってWLCにダウンロードするファイルです。「オプションB:WLC自体によって生成されたCSR」に従った場合、All-certs.pemはWLCにダウンロードする必要があるファイルです。 次の手順では、このファイルを WLC にダウンロードします。

注:SHA2もサポートされています。Cisco Bug ID CSCuf20725 は、SHA512 サポートの要求です。

ステップ 3 CLI：CLI を使用した WLC へのサードパーティ証明書のダウンロード

CLIを使用してチェーン証明書をWLCにダウンロードするには、次の手順を実行します。

1. TFTP サーバ上のデフォルト ディレクトリに final.pem ファイルを移動します。

2. CLIで次のコマンドを入力して、ダウンロード設定を変更します。

```
<#root>

>

transfer download mode tftp

>

transfer download datatype webauthcert

>

transfer download serverip

>

transfer download path
```

```
>

transfer download filename final.pem
```

3. オペレーティング システムで SSL キーと証明書を復号化できるように、.pem ファイルのパスワードを入力します。

<#root>

```
>

transfer download certpassword password
```

注:certpassword の値が、「CSRの生成」セクションのステップ4（または5）で設定した -passout パラメータパスワードと同じであることを確認します。この例では、certpassword の値は check123 でなければなりません。オプションBを選択した場合（つまり、WLC自体を使用してCSRを生成する場合）、certpasswordフィールドは空白のままにします。

---

4. 次を入力します。 **transfer download start** コマンドを発行して、更新された設定を表示します。次に、プロンプトで y と入力して、現在のダウンロード設定を確認し、証明書とキーのダウンロードを開始します。ランダム データの例は次のとおりです。

<#root>

```
(Cisco Controller) >

transfer download start



Mode............................................. TFTP
Data Type........................................ Site Cert
TFTP Server IP................................... 10.77.244.196
TFTP Packet Timeout.............................. 6
TFTP Max Retries................................. 10
TFTP Path......................................../
TFTP Filename.................................... final.pem

This might take some time.
Are you sure you want to start? (y/N)

y
```

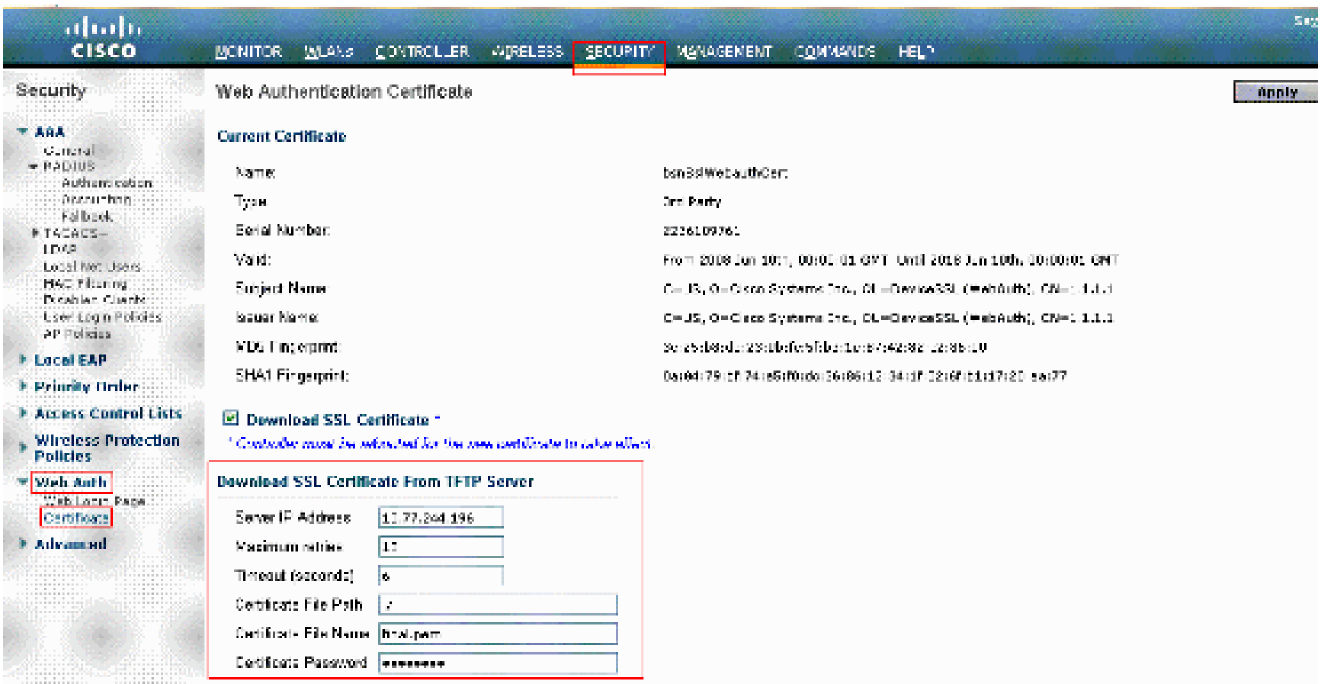
```
TFTP EAP Dev cert transfer start.

Certificate installed.

                        Reboot the switch to use new certificate.
```

5. 変更を有効にするために、WLC をリブートします。

ステップ 3 GUI：GUI を使用した WLC へのサードパーティ証明書のダウンロード

GUIを使用してチェーン証明書をWLCにダウンロードするには、次の手順を実行します。

1. デバイスの証明書 final.pem を TFTP サーバ上のデフォルト ディレクトリにコピーします。
2. 選択 **Security** > **Web Auth** > **Cert** Web Authentication Certificateページを開きます。
3. 次の項目を確認します。 **Download SSL Certificate**チェックボックスをオンにして、Download SSL Certificate From TFTP Serverパラメータを表示します。
4. [IP Address] フィールドに、TFTP サーバの IP アドレスを入力します。



5. [File Path] フィールドに、証明書のディレクトリ パスを入力します。
6. [File Name] フィールドに、証明書の名前を入力します。

7. [Certificate Password] フィールドに、証明書を保護するために使用されたパスワードを入力します。

8. クリック **Apply**を参照。

9. ダウンロードが完了したら、 **Commands** > **Reboot** > **Reboot**を参照。

10. 変更を保存するかどうかを確認するメッセージが表示されたら、 **Save and Reboot**を参照。

11. 変更内容を確定するために [OK] をクリックして、コントローラをリブートします。

トラブルシュート

WLCへの証明書のインストールをトラブルシューティングするには、WLCでコマンドラインを開き、次のように入力します **debug transfer all enable** と **debug pm pki enable** 次に、証明書のダウンロード手順を実行します。

```
In some cases, the logs only say that the certificate installation failed:

*TransferTask: Sep 09 08:37:17.415: RESULT_STRING: TFTP receive complete... Installing
 Certificate.

*TransferTask: Sep 09 08:37:17.415: RESULT_CODE:13


TFTP receive complete... Installing Certificate.

*TransferTask: Sep 09 08:37:21.418: Adding cert (1935 bytes) with certificate key password.

*TransferTask: Sep 09 08:37:21.421: RESULT_STRING: Error installing certificate.
```

証明書の形式とチェーンを確認します。バージョン7.6以降のWLCではチェーン全体が存在する必要があるため、WLC証明書だけをアップロードすることはできません。ルート CA までのチェーンがファイル内に存在する必要があります。

次に、中間 CA が正しくない場合のデバッグの例を示します。

```
*TransferTask: Jan 04 19:08:13.338: Add WebAuth Cert: Adding certificate & private key using password c
*TransferTask: Jan 04 19:08:13.338: Add ID Cert: Adding certificate & private key using password check1
*TransferTask: Jan 04 19:08:13.338: Add Cert to ID Table: Adding certificate (name: bsnSslWebauthCert) 
*TransferTask: Jan 04 19:08:13.338: Add Cert to ID Table: Decoding PEM-encoded Certificate (verify: YES
*TransferTask: Jan 04 19:08:13.338: Decode & Verify PEM Cert: Cert/Key Length was 0, so taking string l
*TransferTask: Jan 04 19:08:13.338: Decode & Verify PEM Cert: Cert/Key Length 7148 & VERIFY
*TransferTask: Jan 04 19:08:13.342: Decode & Verify PEM Cert: X509 Cert Verification return code: 0
*TransferTask: Jan 04 19:08:13.342: Decode & Verify PEM Cert: X509 Cert Verification result text: unabl
*TransferTask: Jan 04 19:08:13.342: Decode & Verify PEM Cert: Error in X509 Cert Verification at 0 dept
```

```
*TransferTask: Jan 04 19:08:13.343: Add Cert to ID Table: Error decoding (verify: YES) PEM certificate
*TransferTask: Jan 04 19:08:13.343: Add ID Cert: Error decoding / adding cert to ID cert table (verifyC
*TransferTask: Jan 04 19:08:13.343: Add WebAuth Cert: Error adding ID cert
```

高可用性（HA SSO）の考慮事項

WLC HA SSO 導入ガイドで説明されているように、HA SSO シナリオでは、証明書がプライマリコントローラからセカンダリコントローラに複製されることはありません。

これは、HAペアを形成する前に、すべての証明書をセカンダリにインポートする必要があることを意味します。

もう1つの注意点は、プライマリWLCでCSRを生成した場合（したがってキーをローカルに作成した場合）は、キーをエクスポートできないため、この機能が動作しないことです。

有効な唯一の方法は、OpenSSL を使用してプライマリ WLC の CSR を生成し（したがって証明書にキーが付属している）、両方の WLC でその証明書/キーの組み合わせをインポートする方法です。

関連情報

- サードパーティ証明書用 CSR の生成とチェーンされていない証明書の WLC へのダウンロード
- Wireless Control System（WCS）でのサードパーティ証明書のための証明書署名要求（CSR）の生成
- Linux サーバ上にインストールされた Wireless Control System（WCS）証明書署名要求（CSR）の設定例
- テクニカル サポートとドキュメント - Cisco Systems
- WLC HA SSO ガイド

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。