Secure Endpoint Private Cloud 3.x以降のインストールに必要な証明書の生成と追加

内容

概要

このドキュメントでは、Secure Console Private Cloudを新規インストールするたびにアップロードする必要がある証明書を生成するプロセス、またはインストールされている証明書サービスを更新するプロセスについて説明します。

前提条件

要件

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。

- Windows Server 2008
- CentOS 7/8
- Secure Console Virtual Private Cloud 3.0.2（以降）
- OpenSSL 1.1.1

使用するコンポーネント

次の項目に関する知識があることが推奨されます。

- Windows Server 2008（以降）
- Secure Console Private Cloudのインストール
- 公開キーインフラストラクチャ
- OpenSSL
- Linux CLI

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、初期（デフォルト）設定の状態から起動しています。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。

背景説明

Secure Console Private Cloud 3.Xの導入に伴い、次のサービスのすべてにホスト名と証明書/キーのペアが必要になります。

- 管理ポータル
- 認証（Private Cloud 3.Xの新機能）
- Secure Console
- Disposition Server
- Disposition Server – 拡張プロトコル
- 廃棄更新サービス
- Firepower Management Center

このドキュメントでは、必要な証明書を簡単に生成してアップロードする方法について説明します。組織のポリシーに従って、ハッシュアルゴリズム、キーサイズなどの各パラメータを微調整できます。また、これらの証明書を生成するメカニズムが、ここで説明する内容と一致しない場合があります。

> **警告**：次に示す手順は、CAサーバの設定によって異なる場合があります。選択したCAサーバがすでにプロビジョニングされ、設定が完了していることが想定されます。次のテクニカルノートでは、証明書を生成する例のみを説明しています。Cisco TACは、証明書の生成に関連する問題や、あらゆる種類のCAサーバの問題のトラブルシューティングには関与していません。

証明書の作成

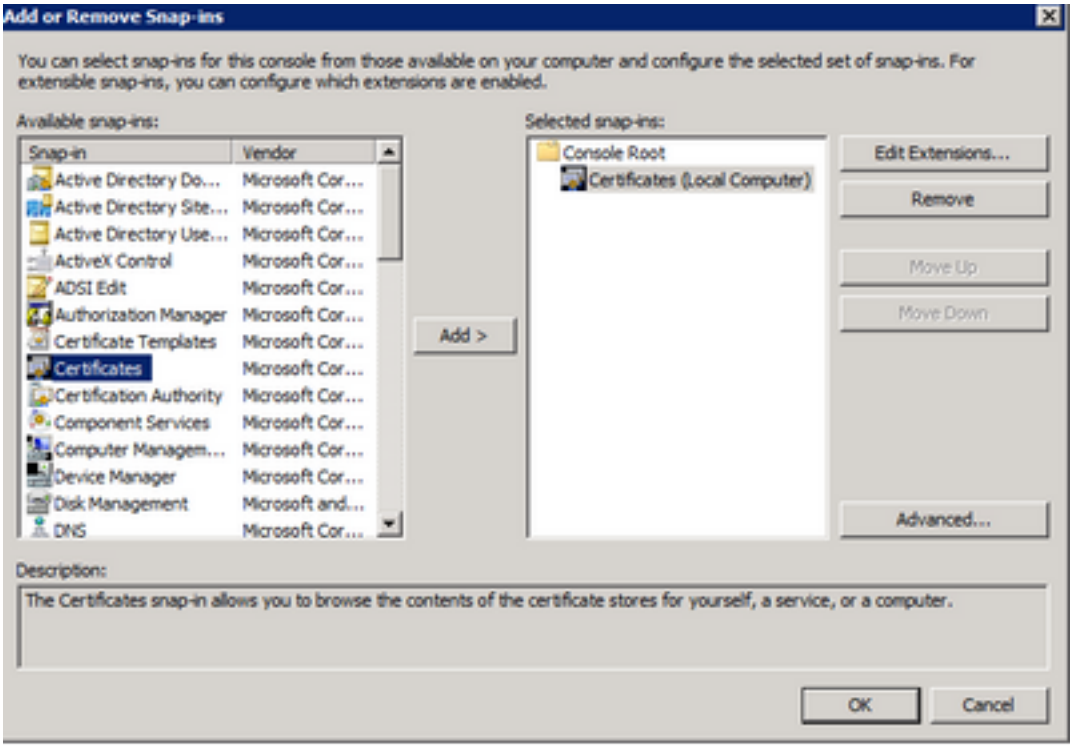
Windowsサーバでの証明書の生成

次の役割がWindows Serverにインストールされ、構成されていることを確認してください。

- Active Directory証明書サービス
- 認証局（CA）
- 証明機関Web登録
- オンラインレスポンダ
- 証明書の登録Webサービス
- 証明書の登録ポリシーWebサービス
- Active Directoryドメインサービス
- DNS Servers
- Webサーバ(IIS)

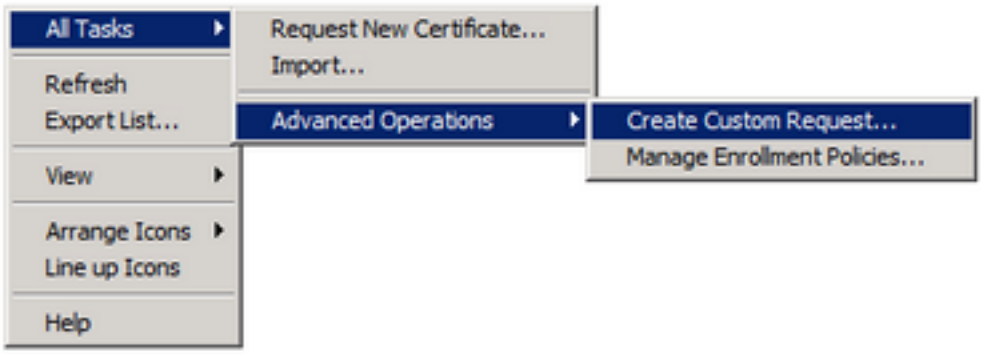
証明書署名要求(CSR)の生成

ステップ 1：MMCコンソールに移動し、次の図に示すように、コンピュータアカウントの証明書スナップインを追加します。

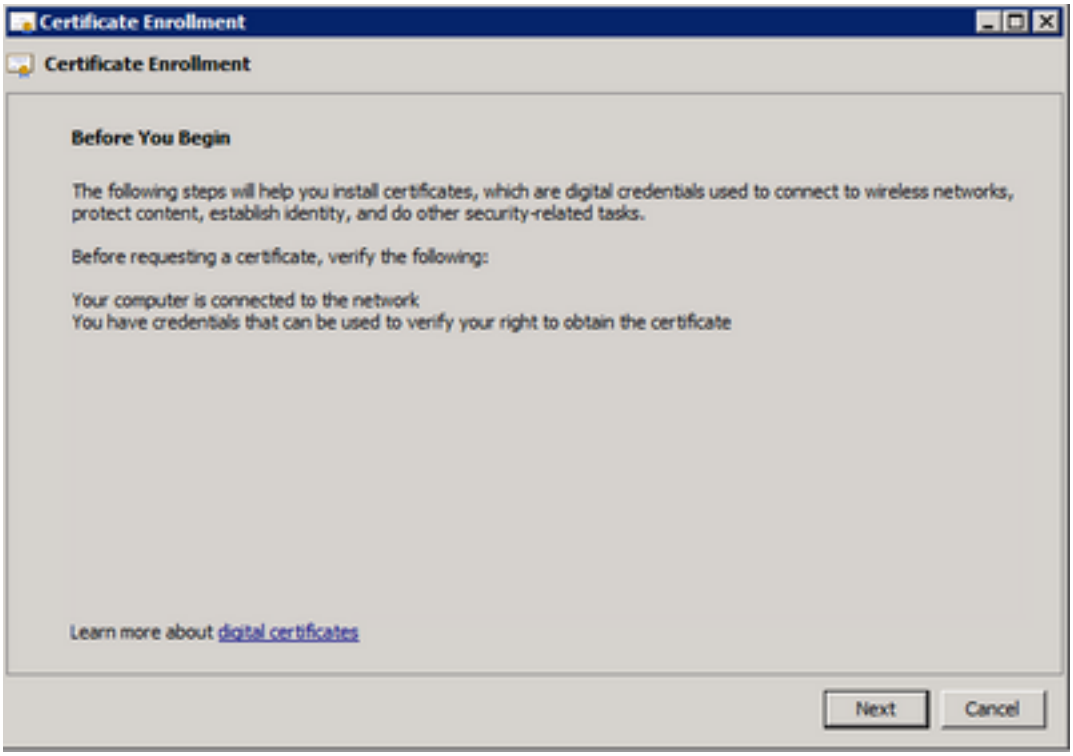


ステップ 2：[Certificates (Local Computer)] > [Personal] > [Certificates] の順にドリルダウンします。

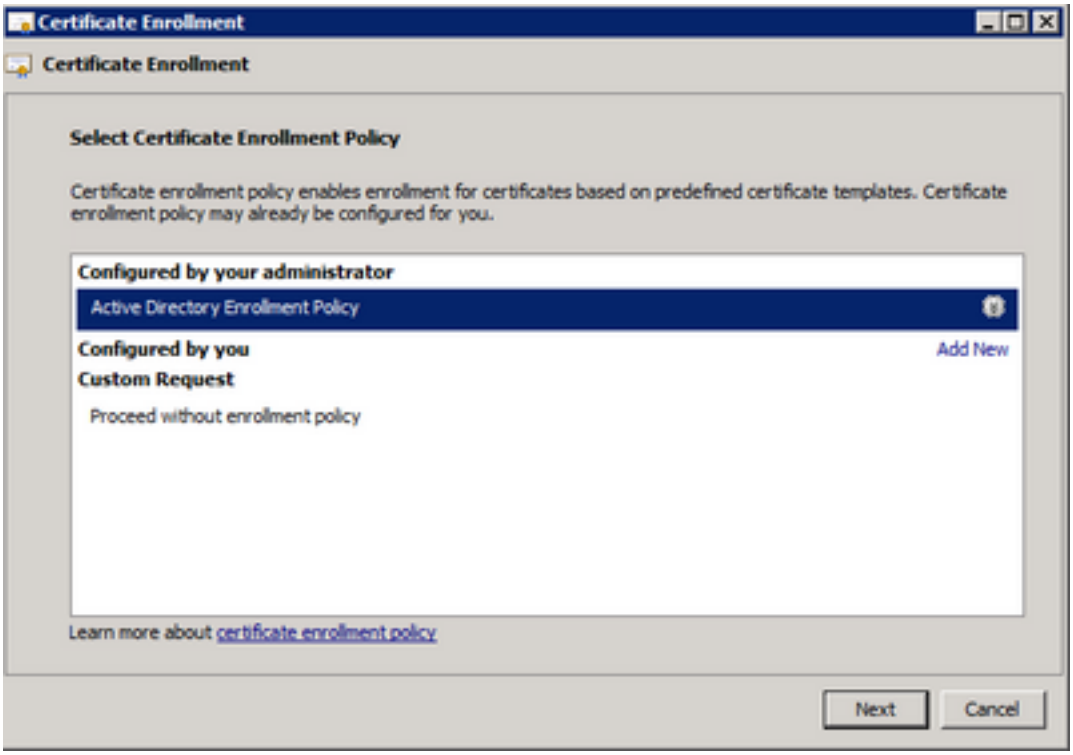
ステップ 3：空きスペースを右クリックし、[すべてのタスク(All Tasks)] > **[高度な操作(Advanced Operations)] > [カスタム要求の作成(Create Custom Request)]を選択します。**



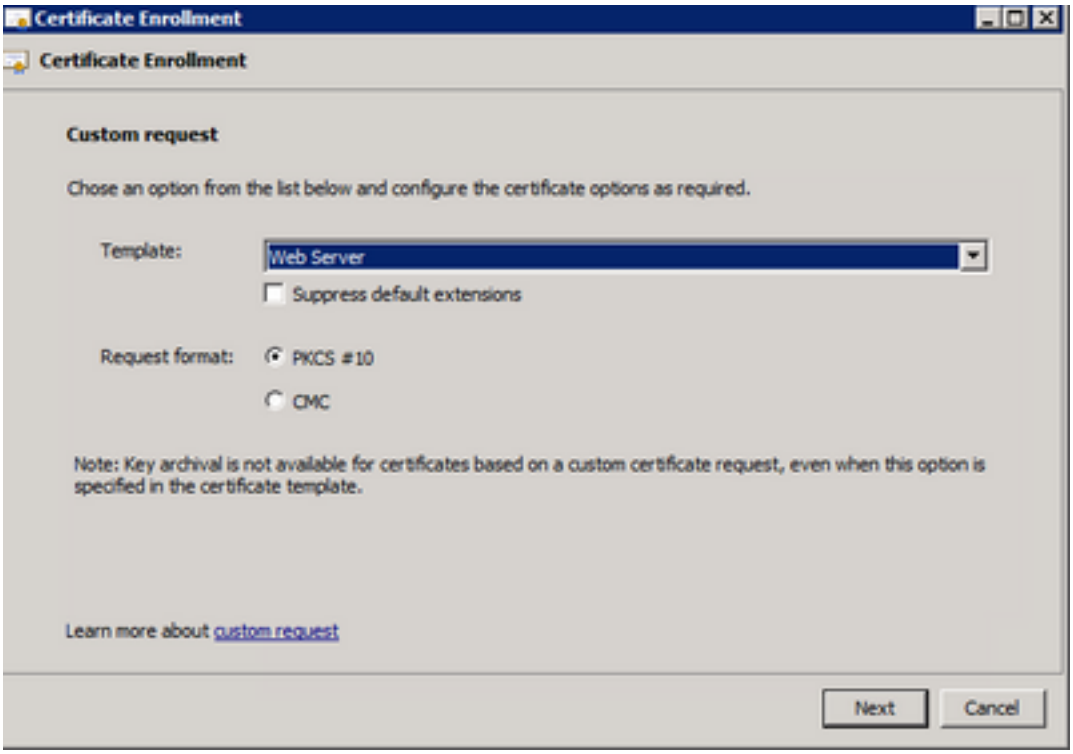
ステップ 4：[Enrollment]ウィンドウで[Next] を選択します。



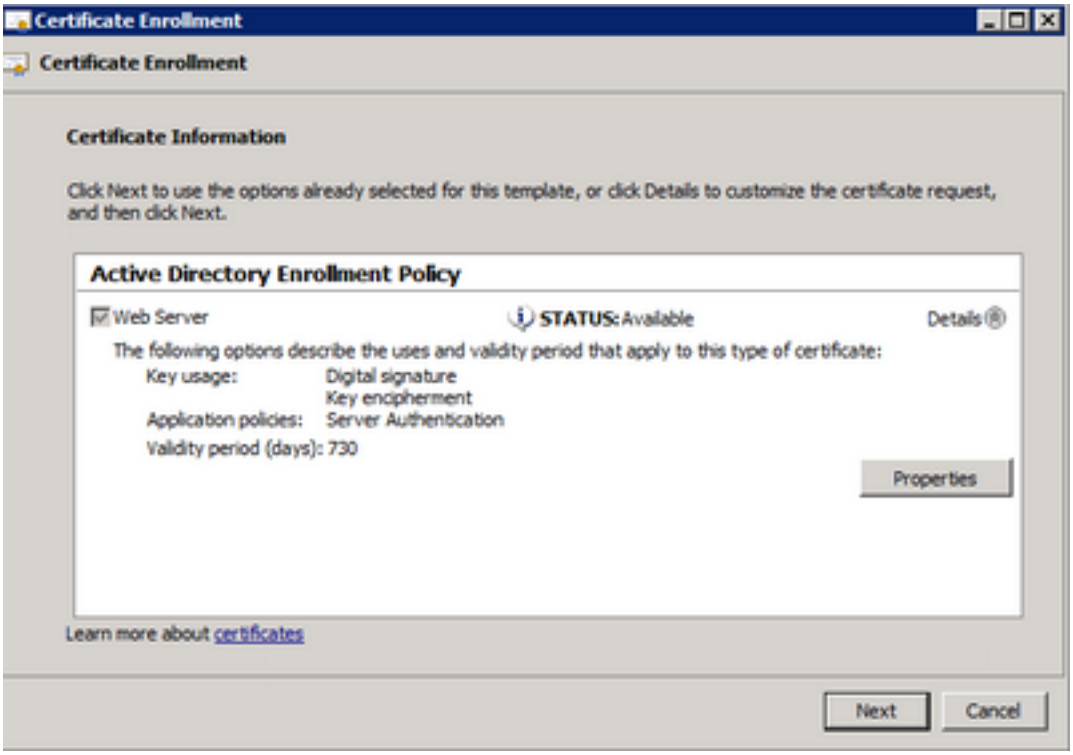
ステップ 5：証明書の登録ポリシーを選択し、[Next] を選択します。



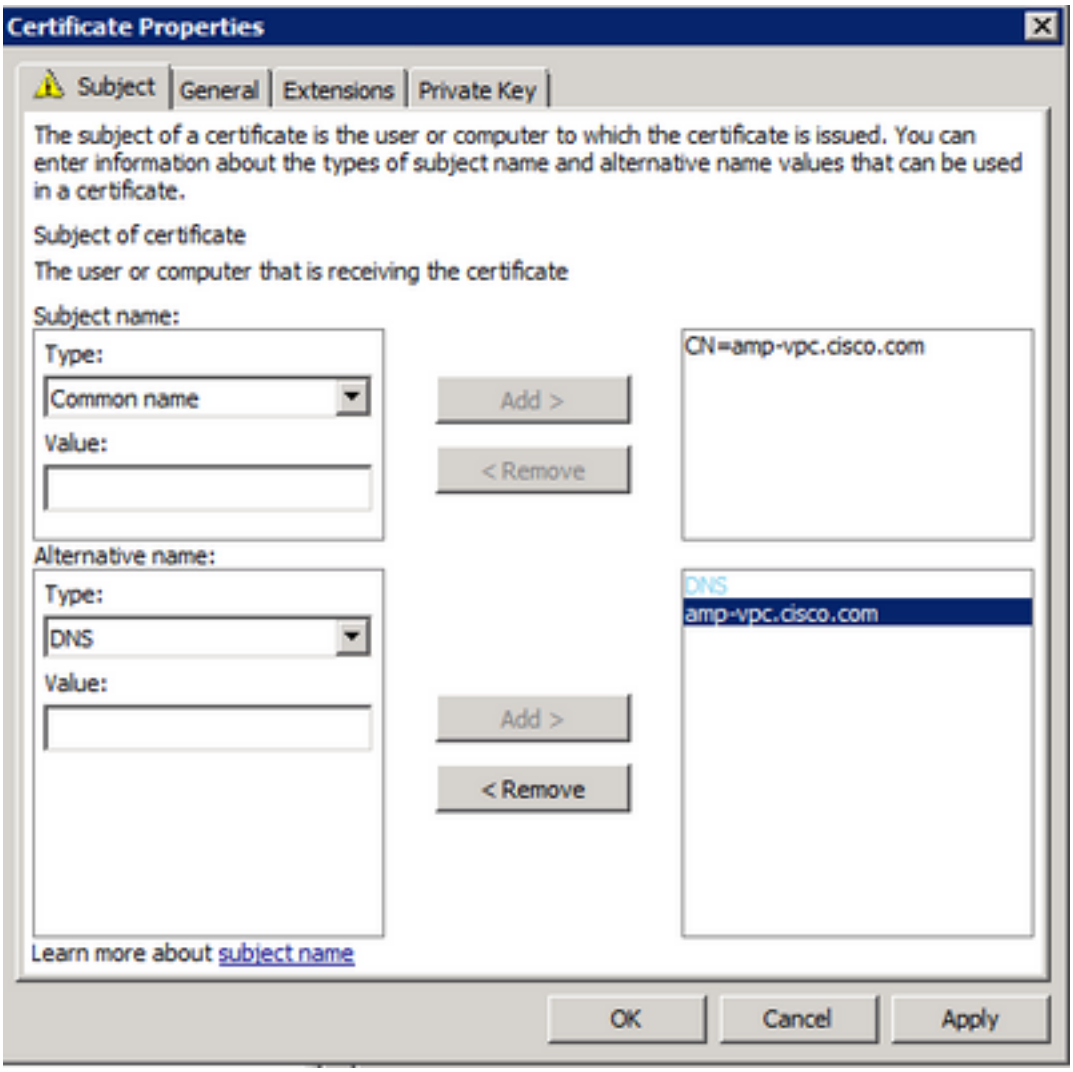
手順 6：テンプレートとして[Web Server] を選択し、[Next] を選択します。



手順 7：「Webサーバー」テンプレートが正しく構成され、登録に使用できる場合は、「使用可能」ステータスが表示されます。**Details**を選択して、Propertiesを展開します。

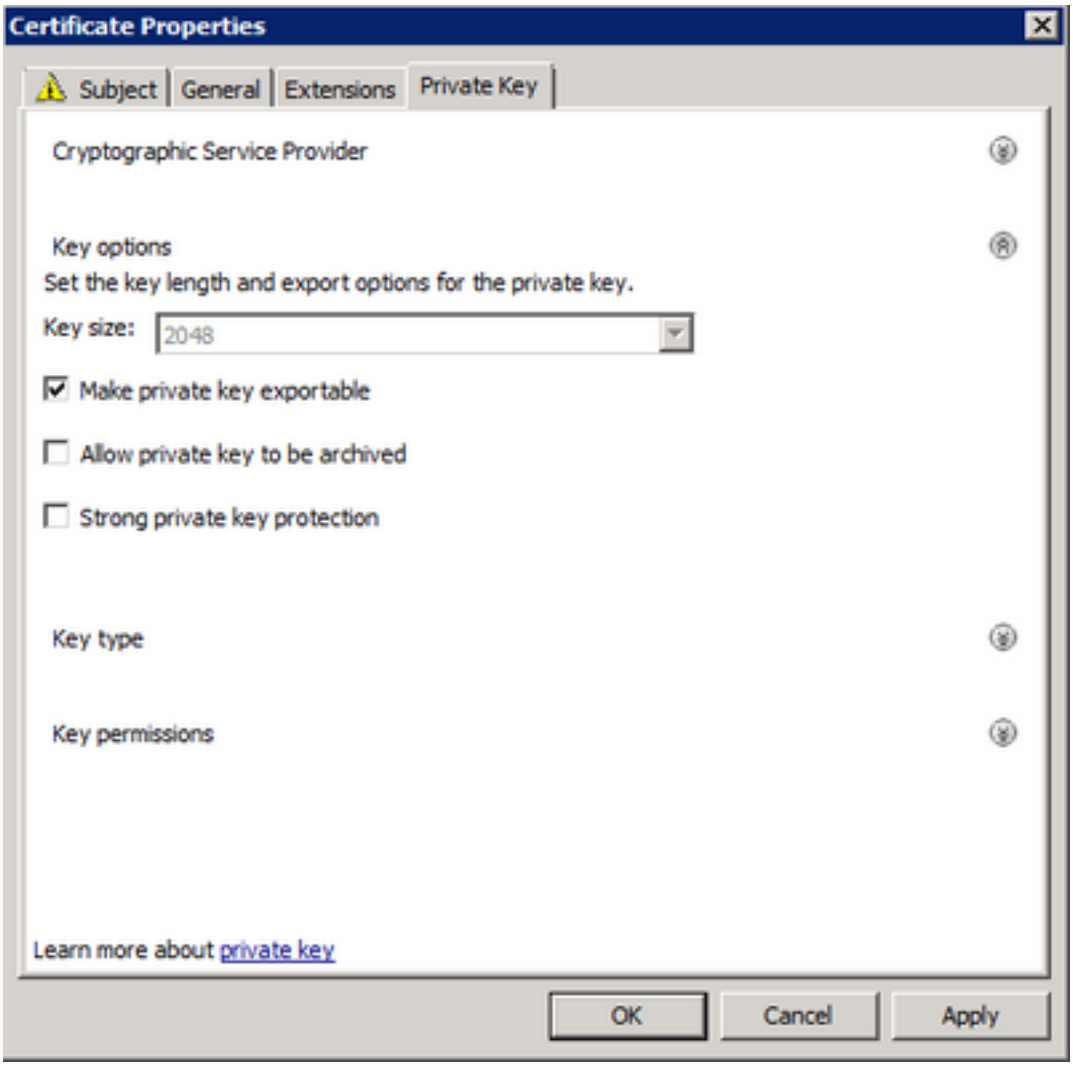


ステップ 8：少なくとも、CN属性とDNS属性を追加します。残りの属性は、セキュリティ要件に従って追加できます。



ステップ 9：必要に応じて、[General] タブの下に[Friendly Name]を入力します。

ステップ 10：[Private **Key]タブを選択し、[Key Options]** セクションで[Make private key exportable] を有効にしていることを確認します。

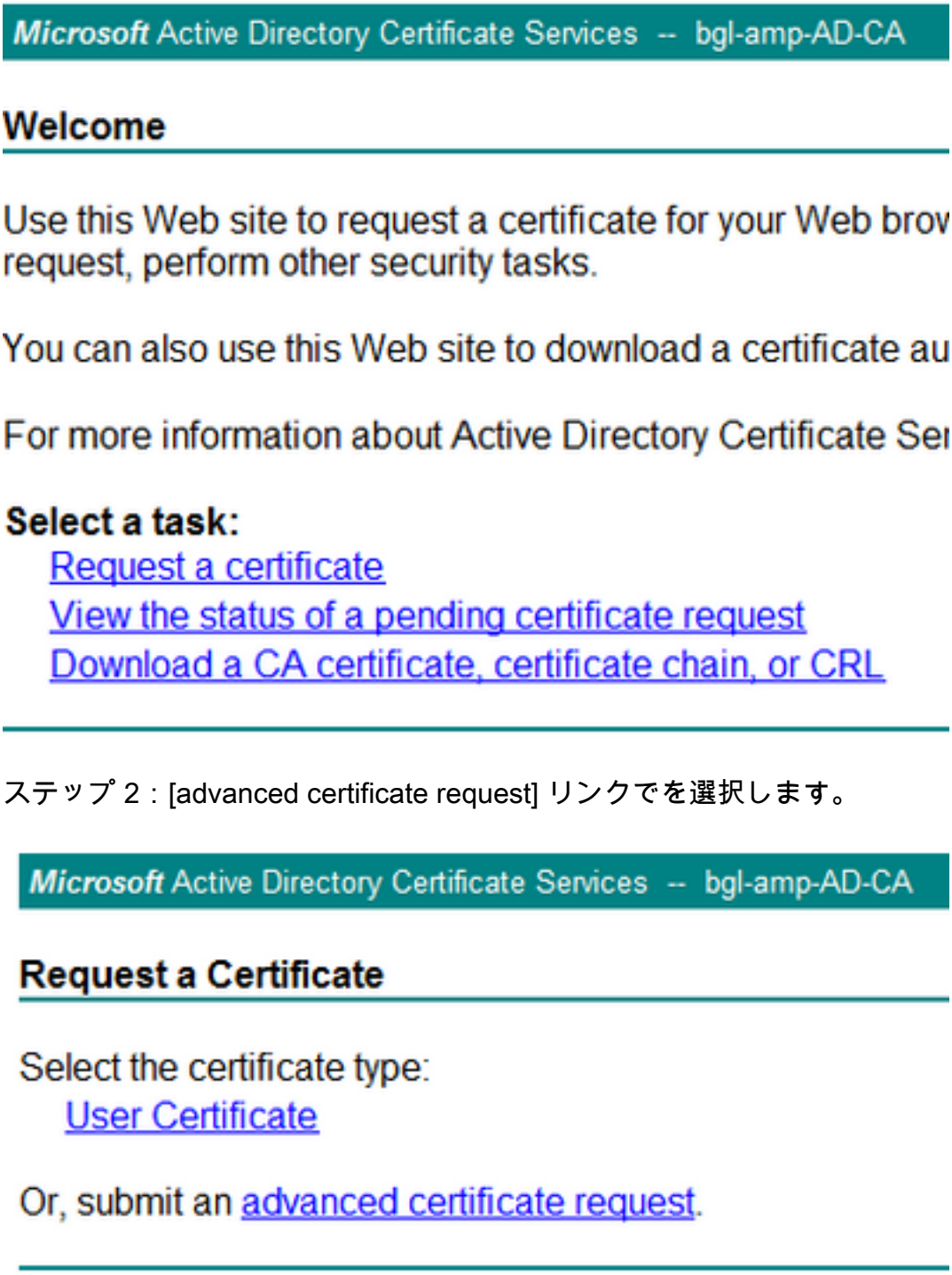


ステップ 11最後に、[OK] を選択します。これにより、[Certificate Enrollment]ダイアログが表示され、[Next] を選択できるようになります。

ステップ 12署名用にCAサーバに送信される.reqファイルを保存する場所を参照します。

CSRをCAに送信し、証明書を生成する

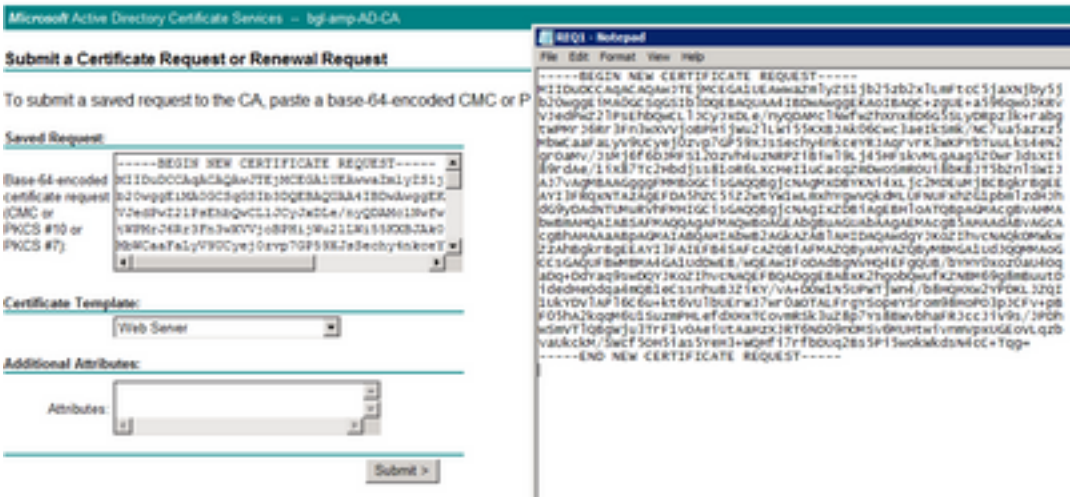
ステップ 1：次のようにMS AD証明書サービスWebページに移動し、[Request a Certificate] を選択します。



ステップ 2：[advanced certificate request] リンクでを選択します。

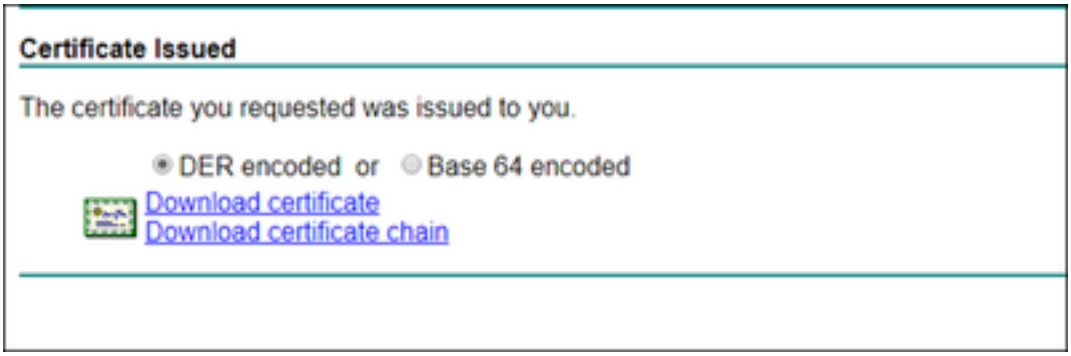
ステップ 3：[Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file, or submit a renewal request by using a base-64-encoded PKCS #7 file] を選択します。

ステップ 4：以前に保存した.reqファイル(CSR)の内容をメモ帳で開きます。内容をコピーして、ここに貼り付けます。[Certificate Template]が[Web Server] として選択されていることを確認します



ステップ 5：最後に、**Submit**を選択します。

手順 6：この時点で、次の図に示すように、証明書を**ダウンロード**できる必要があります。



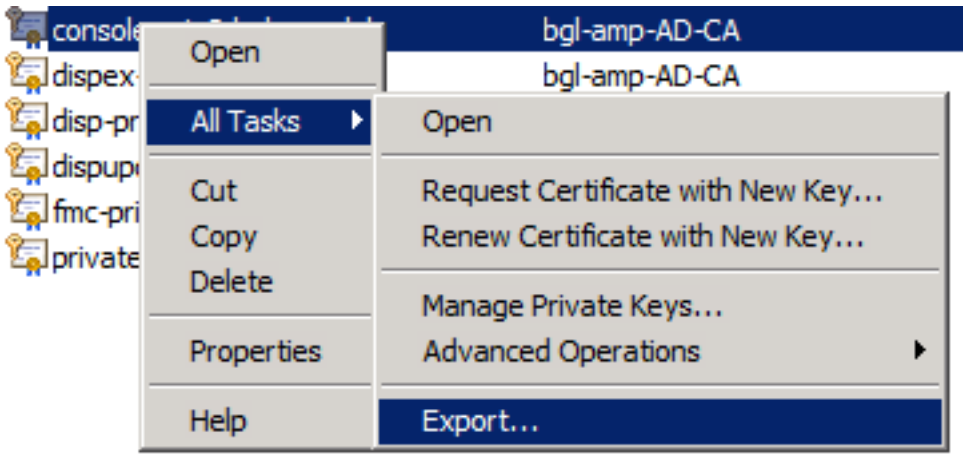
秘密キーのエクスポートとPEM形式への変換

ステップ 1：.cerファイルを開いて証明書を証明書ストアにインストールし、[Install Certificate] を選択します。

ステップ 2：先ほど選択したMMCスナップインに移動します。

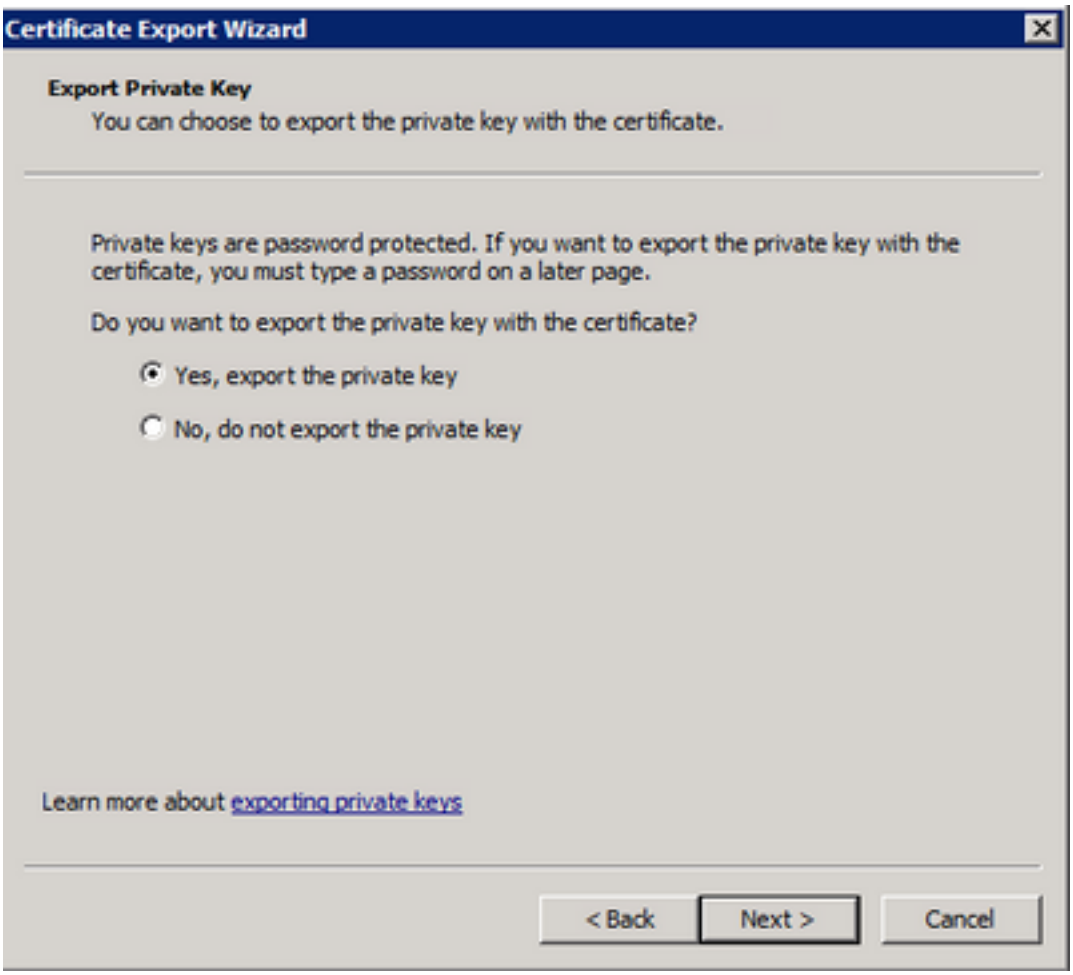
ステップ 3：証明書がインストールされたストアに移動します。

ステップ 4：正しい証明書を右クリックし、[All Tasks] > [Export] を選択します。



ステップ 5：[Certificate Export Wizard]で、次の図に示すように、秘密キーをエクスポートするこ

とを確認します。



手順 6：パスワードを入力し、[Next] を選択して秘密キーをディスクに保存します。

手順 7：これにより、秘密キーは.PFX形式で保存されますが、Secure Endpoint Private Cloudで使用するには、.PEM形式に変換する必要があります。

ステップ 8：OpenSSLライブラリをインストールします。

ステップ 9：コマンドプロンプトウィンドウを開き、OpenSSLをインストールしたディレクトリに移動します。

ステップ 10：次のコマンドを実行して秘密キーを抽出し、新しいファイルに保存します。（PFXファイルがOpenSSLライブラリと同じパスにない場合は、ファイル名と共に正確なパスを指定する必要があります）

```
openssl pkcs12 -in yourpfxfile.pfx -nocerts -out privatekey.pem -nodes
```

ステップ 11次のコマンドを実行して、公開証明書も抽出し、新しいファイルに保存します。

```
openssl pkcs12 -in yourpfxfile.pfx -nokeys -out publiccert.pem -nodes
```

Linuxサーバでの証明書の生成（厳密なSSLチェックは無効）

注:厳密なTLSチェックでは、証明書がAppleのTLS要件を満たしているかどうかを確認します。詳細については、『管理者ガイド』を参照してください。

必要な証明書を生成しようとしているLinuxサーバにOpenSSL 1.1.1ライブラリがインストールされていることを確認します。この問題と次に示す手順が、実行しているLinuxディストリビューションと異なる可能性があるかどうかを確認します。この部分は、CentOS 8.4サーバで行われたように文書化されています。

自己署名ルートCAの生成

ステップ 1：ルートCA証明書の秘密キーを生成します。

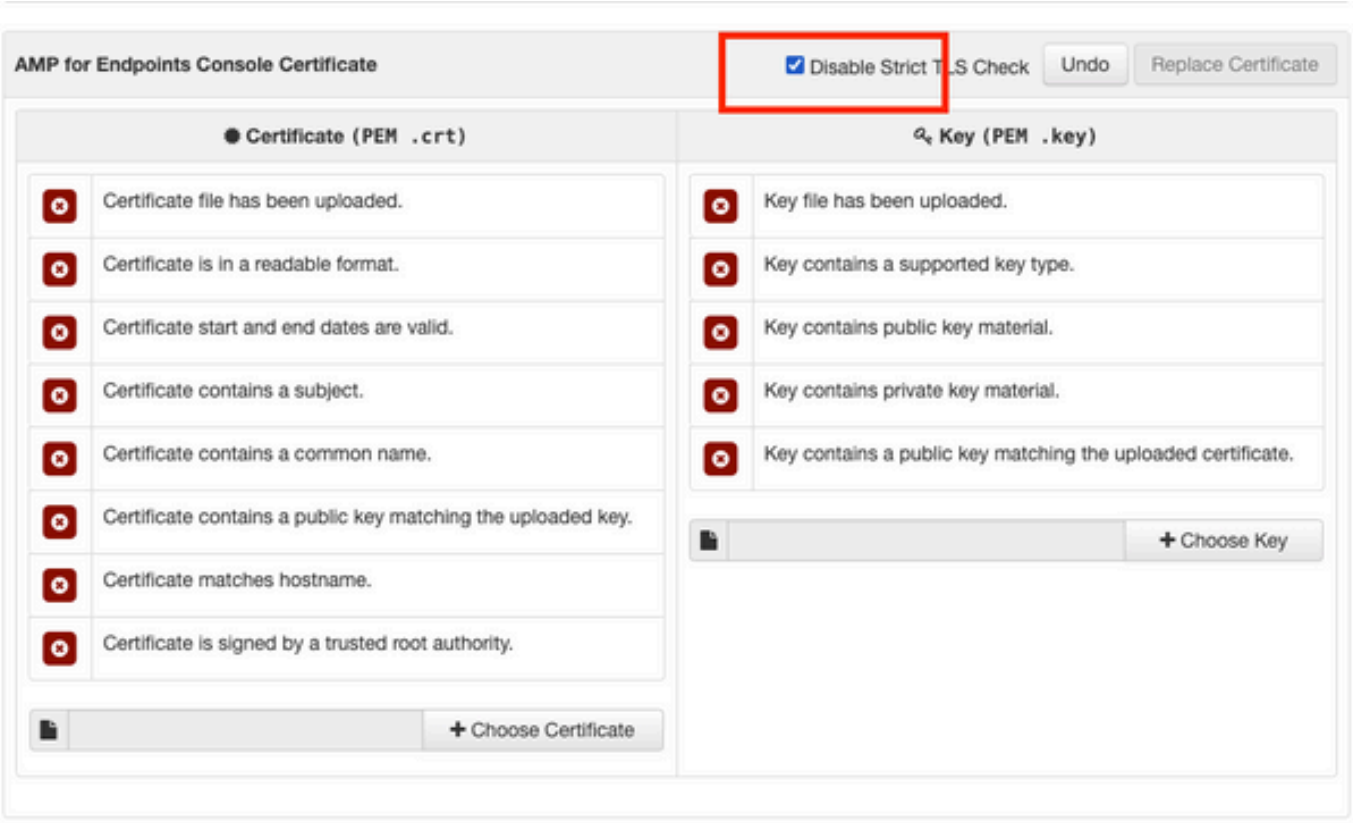
```
openssl genrsa -out
```

ステップ 2：CA証明書を生成します。

```
openssl req \
-subj '/CN=
-addext "extendedKeyUsage = serverAuth, clientAuth" \
-outform pem -out
-key
-days "1000"
```

各サービスの証明書を生成する

DNS名エントリに従って、認証、コンソール、ディスポジション、ディスポジション拡張、アップデートサーバ、Firepower Management Center(FMC)サービスの証明書を作成します。サービス (認証、コンソールなど) ごとに、次の証明書生成プロセスを繰り返す必要があります。



秘密キーの生成

```
openssl genrsa -out
```

<YourServiceName.key>を、Auth-Cert.keyとして作成される新しいKEYファイル名に置き換えます

CSR の生成

```
openssl req -new \
-subj '/CN=
-key
```

Cisco IOSソフトウェアリリース12.1 Auth-Cert.keyなどの現在（または新しい）証明書KEYファイルを使用した<YourServiceName.key>

<YourServiceName.csr>を、作成するCSRファイル名（Auth-Cert.crtなど）に置き換えます

証明書の生成

```
openssl x509 -req \
-in
-CAkey
-days 397 -sha256
```

<YourServiceName.csr>をAuth-Cert.csrなどの実際の（または新しい）証明書CSRに置き換えます

<YourRootCAName.pem>をRootCAName.pemという実際の（または新しい）PEMファイル名に置き換えます。

<YourServiceName.key>をAuth-Cert.keyなどの現在（または新しい）証明書KEYファイルに置き換えます

<YourServiceName.crt>をAuth-Cert.crtなどの作成するファイル名に置き換えます

Linuxサーバでの証明書の生成（厳密なSSLチェックが有効）

> **注:厳密なTLSチェックでは、証明書がAppleのTLS要件を満たしているかどうかを確認します。**詳細については、『管理者ガイド』を参照してください。

自己署名ルートCAの生成

ステップ 1：ルートCA証明書の秘密キーを生成します。

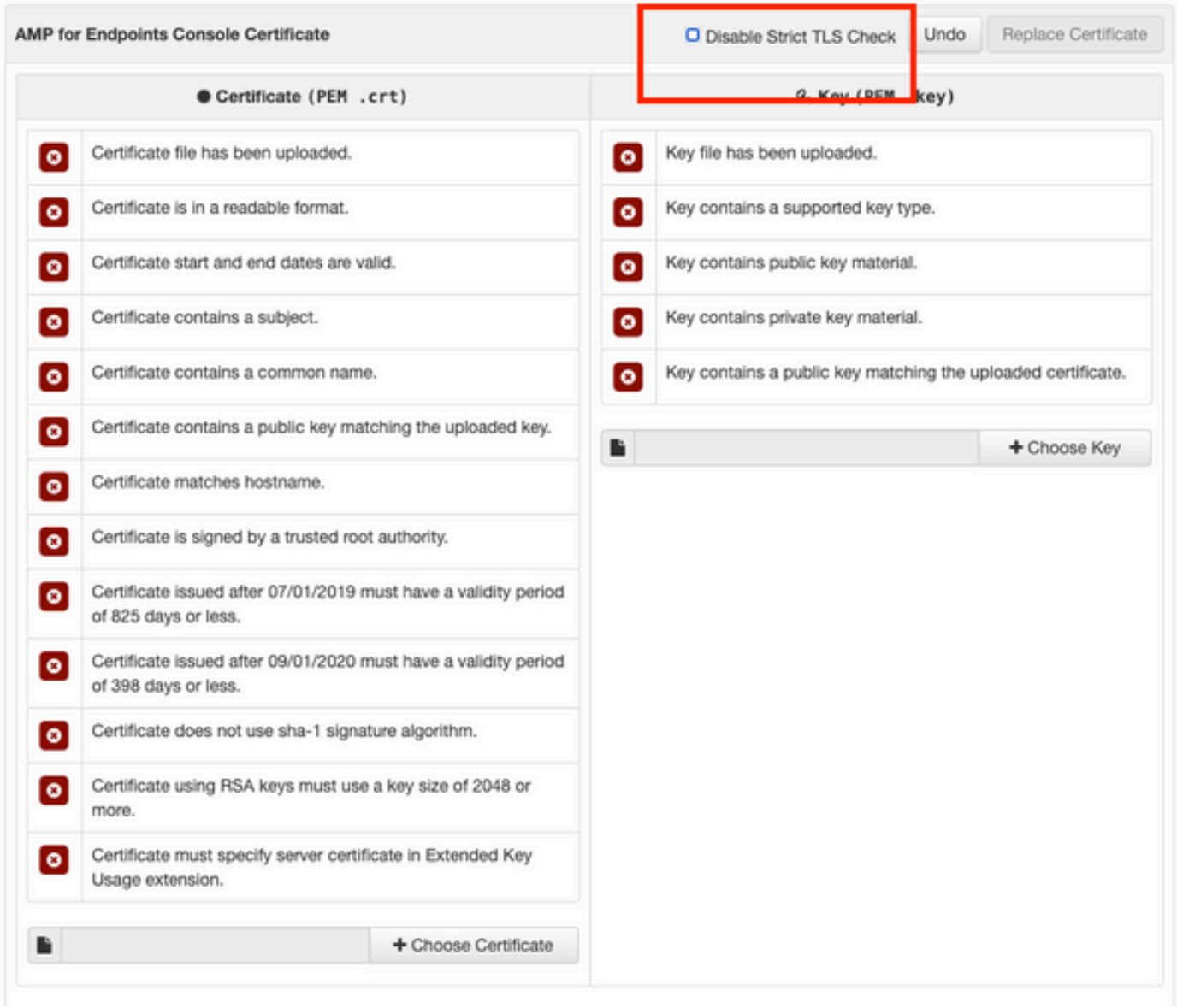
```
openssl genrsa -out
```

ステップ 2：CA証明書を生成します。

```
openssl req \
-subj '/CN=
-outform pem -out
-key
-days "1000"
```

各サービスの証明書を生成する

DNS名エントリに従って、認証、コンソール、ディスポジション、ディスポジション拡張、アップデートサーバ、Firepower Management Center(FMC)サービスの証明書を作成します。サービス（認証、コンソールなど）ごとに、次の証明書生成プロセスを繰り返す必要があります。



Extensions Configurationファイルを作成して保存します(extensions.cnf)。

```
[v3_ca]
basicConstraints = CA:FALSE
keyUsage = critical, digitalSignature, keyEncipherment
extendedKeyUsage = critical, serverAuth, clientAuth
```

秘密キーの生成

```
openssl genrsa -out
```
<YourServiceName.key>を、Auth-Cert.keyとして作成される新しいKEYファイル名に置き換えます

CSR の生成

```
openssl req -new \
-key
-subj '/CN=
-out
```
Cisco IOSソフトウェアリリース12.1 Auth-Cert.keyなどの現在（または新しい）証明書KEYを使用した<YourServiceName.key>

<YourServiceName.csr>を、Auth-Cert.csrなどの現在の（または新しい）証明書CSRに置き換えます

証明書の生成

```
openssl x509 -req -in
-CA
-CAcreateserial -out
-extensions v3_ca -extfile extensions.cnf \
-days 397 -sha256
```
<YourServiceName.csr>をAuth-Cert.csrなどの現在（または新しい）証明書CSRに置き換えます

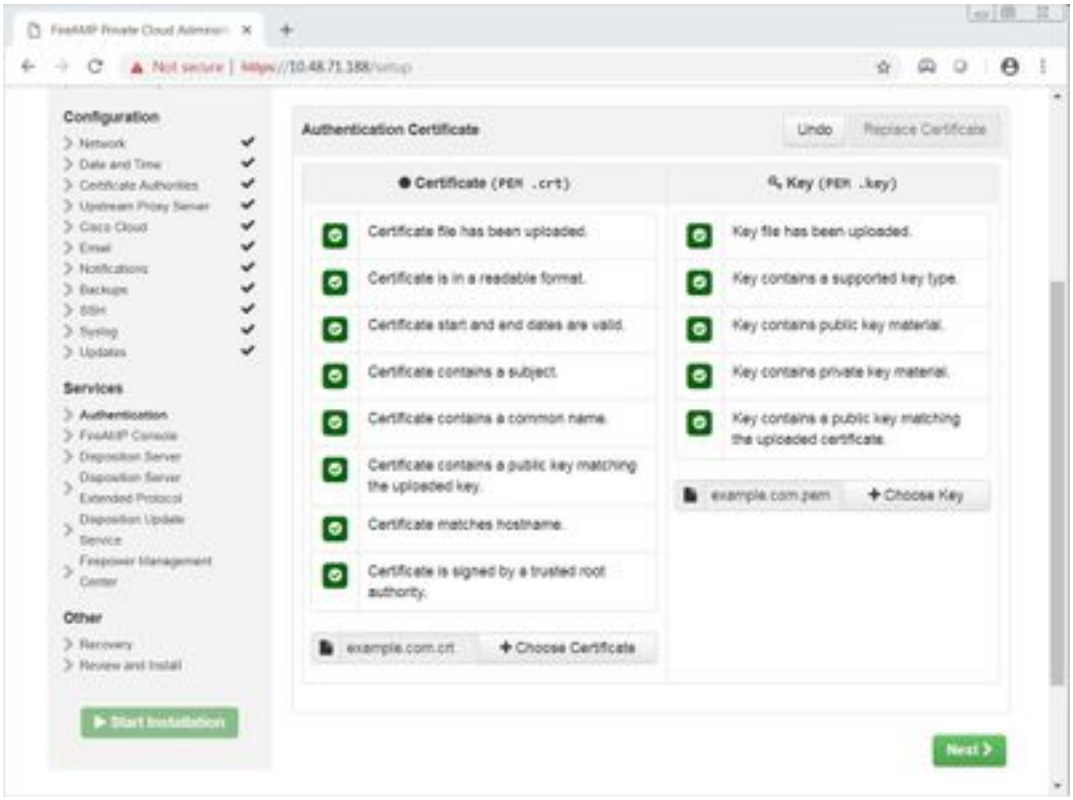
<YourRootCAName.pem>をRootCAName.pemとして現在（または新しい）PEMファイル名に置き換えます。

<YourServiceName.key>をAuth-Cert.keyなどの現在（または新しい）証明書KEYファイルに置き換えます

<YourServiceName.crt>をAuth-Cert.crtなどの作成するファイル名に置き換えます

Secure Console Private Cloudへの証明書の追加

ステップ 1：上記のいずれかの方法で証明書が生成されたら、各サービスに対応する証明書をアップロードします。正しく生成されている場合は、次の図に示すように、すべてのチェックマークが有効になります。



確認

現在、この設定に使用できる確認手順はありません。

トラブルシュート

現在、この設定に関する特定のトラブルシューティング情報はありません。

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。