複数の URL の自己署名証明書の作成

目次

概要

この資料に複数の URL と CloudCenter によって使用できる自己署名証明書を作成する方法を記述されています。

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。

- 証明書
- Linux

使用するコンポーネント

この文書に記載されている情報は CentOS7 に基づいています。

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たものです。 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。 ネットワークが稼働中の場合は、コマンドが及ぼす潜在的な影響を十分に理解しておく必要があります。

問題

規格 CloudCenter の来るか、または Cisco Call Manager （CCM） コンフィギュレーション ウィザードの使用で作成することができる認証にある特定のブラウザが、Google Chrome のような工ラーとして、扱い、警告する認証対象代替名前（SAN）がありません。 これは無効にすることができます SAN なしで、認証は 1 つの特定の URL から有効であるただ場合もあります。

たとえば、www.opencart.com が 10.11.12.13）に属するものとしてホスト ファイルにリストされていてもその URL が認証がであるもののためではないので www.opencart.com のドメイン ネーム システム（DNS）名があれば 10.11.12.13 の IP アドレスのために有効、受け取る Certificate エラーをである認証がある場合（これは本当です。 これは各 SSO サーバに自身の URL があるので CloudCenter の転借人が単一 サインの使用に（SSO）ついている場合予期せず起こることがで

きます。

解決策

この問題を解決する最も簡単な方法は同じ IP アドレスに指示する URL をリストする SAN がある新しい自己署名証明書を作成することです。 ガイドはこのプロセスに最良 の 方法を適用する試みです。

ステップ 1.**ルート ディレクトリ**にナビゲート し、認証を収容するために新しいフォルダーを作って下さい:

```
sudo -s
cd /root
mkdir ca
```

ステップ 2.新しいフォルダーへのナビゲートは認証、プライベートキーおよびログを編成するためにサブ フォルダを作り。

```
cd ca
mkdir certs crl newcerts private
chmod 700 private
touch index.txt
echo 1000 > serial
```

ステップ 3. **/root/ca/openssl.cnf に CAopenssl.conf** のコンテンツをコピーして下さい

> **注:** このファイルは CloudCenter のために適切であるかもしれない省略時 の オプションおよび認証局 （ CA ）のための設定 オプションが含まれています。

ステップ 4. CA のためのプライベートキーおよび認証を生成して下さい。

```
openssl genrsa -aes256 -out private/ca.key.pem 4096
chmod 400 private/ca.key.pem
openssl req -config openssl.cnf -key private/ca.key.pem -new -x509 -days 7300 -sha256 -
extensions v3_ca -out certs/ca.cert.pem
chmod 444 certs/ca.cert.pem
```

ステップ 5 CA は不正 なユーザーによってどの認証でも有効であることを確認する最終的な方法、この認証決してアクセスされ、はインターネットに決して露出されてはなりますではないです。 中間機関認証がそれ妥協される取り消すことができるおよび発行される新しいものをなりませんこの制約事項が原因で、これ作成します中断端認証に署名する中間 CA を作成しなければ。

ステップ 6 中間 CA のための新しいサブディレクトリを作って下さい。

```
mkdir /root/ca/intermediate
cd /root/ca/intermediate/
mkdir certs crl csr newcerts private
chmod 700 private
touch index.txt
echo 1000 > serial
echo 1000 > /root/ca/intermediate/crlnumber
```

ステップ 7. /root/ca/intermediate/openssl.cnf **に Intermediateopenssl.conf** のコンテンツをコピーして下さい。

> **注:** このファイルは少数の小さい微調整以外 CA のためのほぼ同一の設定 オプションが中間物に特定にする含まれています。

ステップ 8.中間キーおよび認証を生成して下さい。

```
cd /root/ca
openssl genrsa -aes256 -out intermediate/private/intermediate.key.pem 4096
chmod 400 intermediate/private/intermediate.key.pem
openssl req -config intermediate/openssl.cf -new -sha256 -key
intermediate/private/intermediate.key.pem -out intermediate/csr/intermediate.csr.pem
```

ステップ 9. CA 認証の中間認証に、これ構築します認証の信頼性を確認するのにブラウザが使用する信頼のチェーンを署名して下さい。

```
openssl ca -config openssl.cnf -extensions v3_intermediate_ca -days 3650 -notext -md sha256 -in
intermediate/csr/intermediate.csr.pem -out intermediate/certs/intermediate.cert.pem
chmod 444 intermediate/certs/intermediate.cert.pem
```

ステップ 10.インターネットの CA がほしいと思わないので CA チェーンを、CA まで信頼性をずっと確認するのにブラウザが使用する CA チェーンを作ることができます作成して下さい。

```
cat intermediate/certs/intermediate.cert.pem certs/ca.cert.pem > intermediate/certs/ca-
chain.cert.pem
chmod 444 intermediate/certs/ca-chain.cert.pem
```

ステップ 11. CCM のための New 鍵および認証を作成して下さい。

```
openssl genrsa -out intermediate/private/ccm.com.key.pem 2048
openssl req -new -sha256 -key intermediate/private/ccm.com.key.pem -subj
"/C=US/ST=NC/O=Cisco/CN=ccm.com" -reqexts SAN -config <(cat intermediate/openssl.cnf <(printf
"[SAN]\nsubjectAltName=DNS:ccm.com,DNS:www.ccm.com,IP:10.11.12.13")) -out
intermediate/csr/ccm.com.csr
```

手順 12： これにコマンドですべての必要フィールドがあり、手動で編集されなければなりません。

- /C =US は示します国（2 つの文字制限）を
- /ST =NC は状態を示し、領域を含むかもしれません
- /O =Cisco は組織を示します
- /CN =ccm.com は Common Name を、これ CCM にアクセスするのに使用される主要な URL であるはずです示します。
- SAN \ nsubjectAltName= は代替名です、Common Name はこのリストであるはずで、SAN のあるか何かへの制限がありません。

手順 13： 中間認証の使用の最終的な認証に署名して下さい。

```
openssl ca -config intermediate/openssl.cnf -extensions server_cert -days 375 -notext -md sha256
-in intermediate/csr/ccm.com.csr -out intermediate/certs/ccm.com.cert.pem
```

手順 14： 認証が正しく署名したことを確認して下さい。

```
openssl verify -CAfile intermediate/certs/ca-chain.cert.pem intermediate/certs/ccm.com.cert.pem
```

手順 15： それは OK か失敗を戻すことができます。

ステップ 16。 新しい認証をコピーして下さい、それは Catalina フォルダへのキーおよび CA チェーンです。

```
cd /root/ca/intermediate/certs
cp ccm.com.cert.pem /usr/local/tomcat/conf/ssl/ccm.com.crt
cp ca-chain.cert.pem /usr/local/tomcat/conf/ssl/ca-chain.crt
cd ../private
cp ccm.com.key.pem /usr/local/tomcat/conf/ssl/ccm.com.key
```

ステップ 17。 アクセス許可 cliqruser 所有権および正しのセット権限。

```
chown cliqruser:cliqruser ccm.com.crt
chown cliqruser:cliqruser ccm.com.key
chown cliqruser:cliqruser ca-chain.crt
chmod 644 ccm.com.crt
chmod 644 ccm.com.key
chmod 644 ca-chain.crt
```

ステップ 18。 変更を行なう前に **server.xml ファイルを**バックアップして下さい。

```
cd ..
cp server.xml server.xml.bak
```

ステップ 19。 **server.xml を**編集して下さい:

1. セクションを見つけて下さい **<Connector port="10443" maxHttpHeaderSize="8192"** と開始する
2. ccm.com.crt を指すために **SSLCertificateFile を**変更して下さい
3. ccm.com.key を指すために **SSLCertificateKeyFile を**変更して下さい
4. カリフォル二アchain.crt を指すために **SSLCACertificateFile を**変更して下さい

ステップ 20。 再始動 Tomcat。

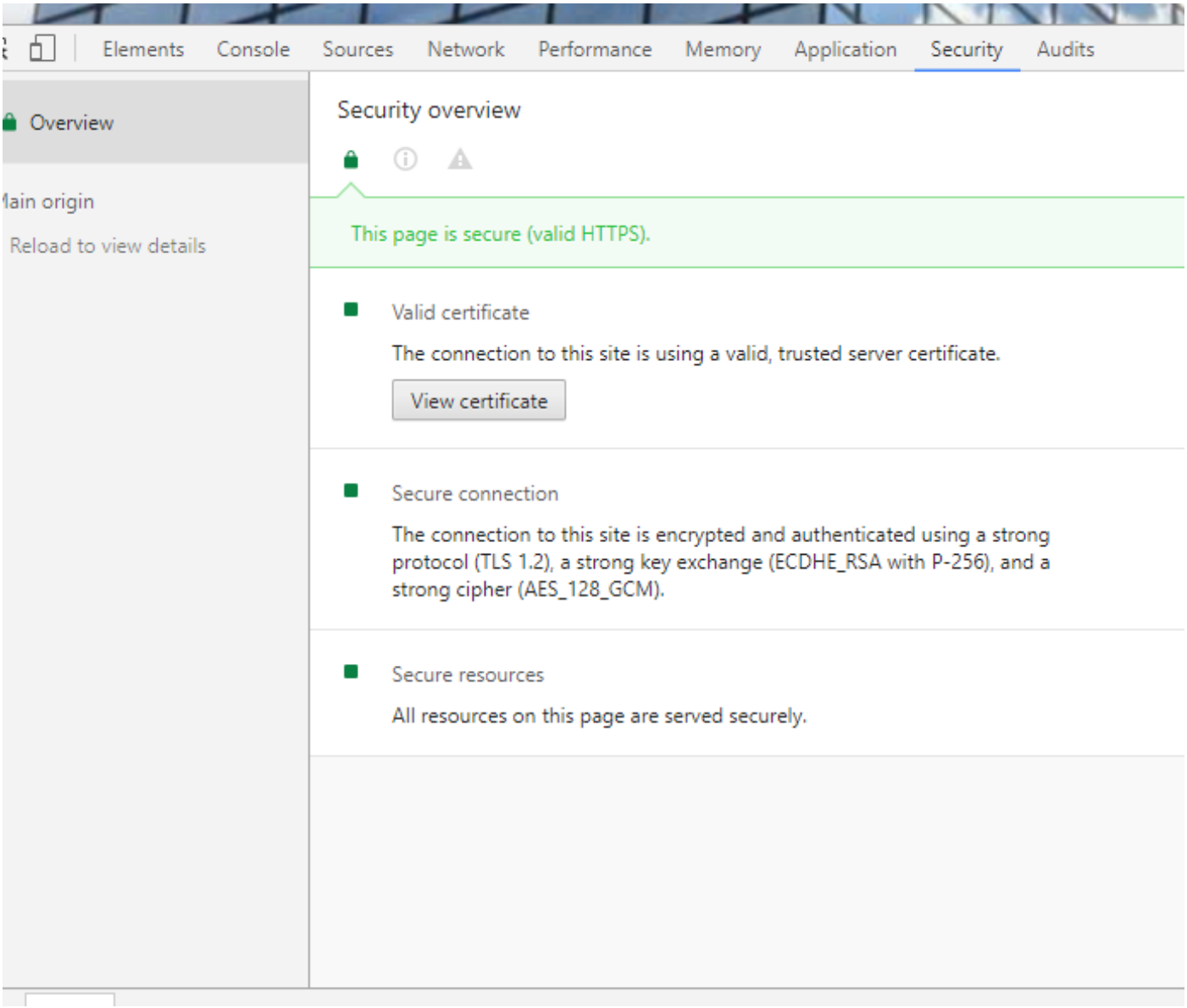
```
service tomcat stop
service tomcat start
```

ステップ 21。 CCM はステップ 13 で規定 されるすべての DNS名および IP アドレスのために有効である新しい認証を使用するようになります。

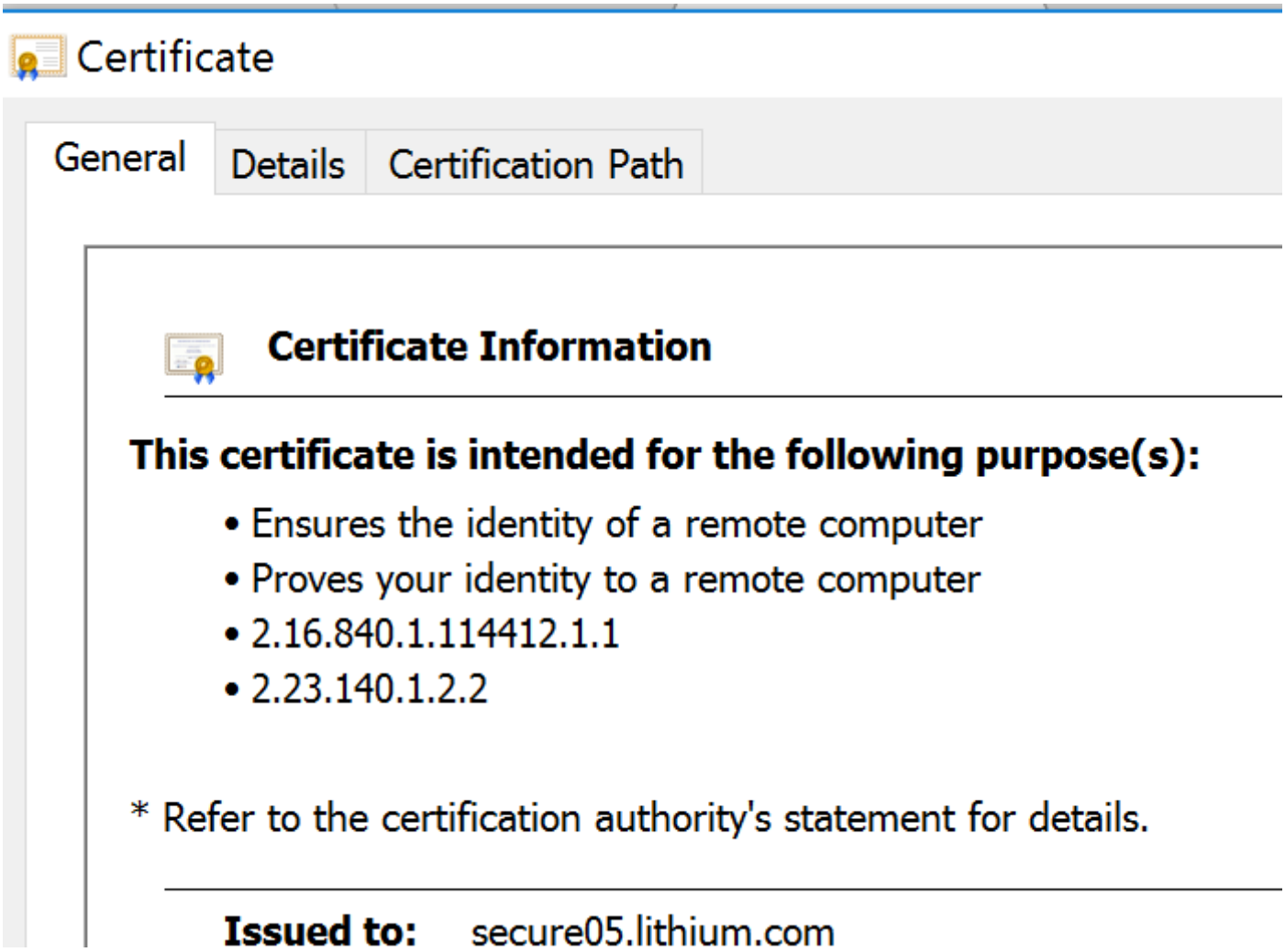
ステップ 22。 CA がガイドの時に作成されたので、ブラウザは有効なとしてそれを、手動で 認証をインポートしなければなりませんデフォルトで認識しません。

ステップ 23。 あらゆる有効な URL の使用の **CCM** へのナビゲートは **Ctrl+Shift+i を**、これ開きます開発者ツールを押し。

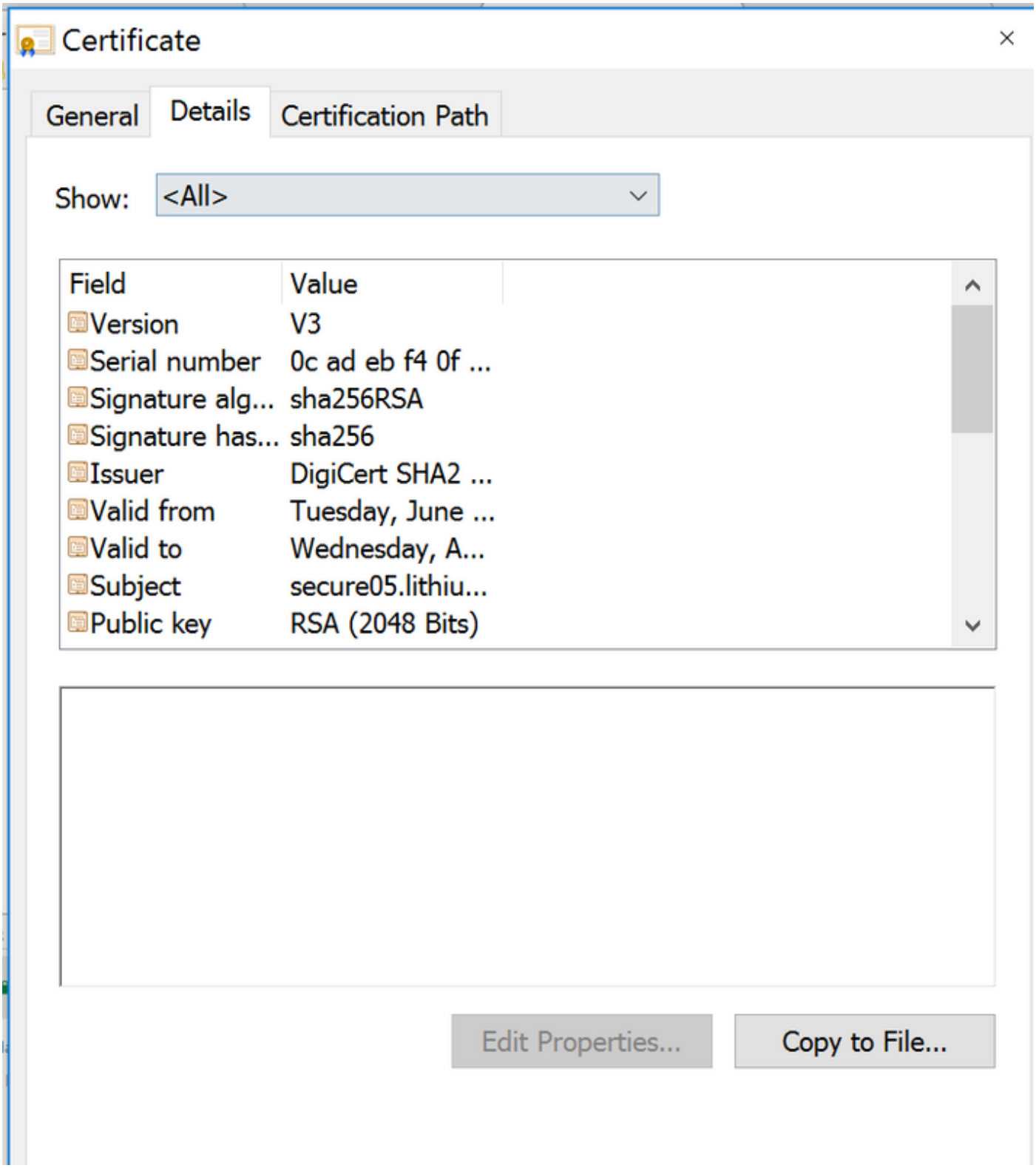
ステップ 24。 イメージに示すように**認証を**『View』 を選択 して下さい。



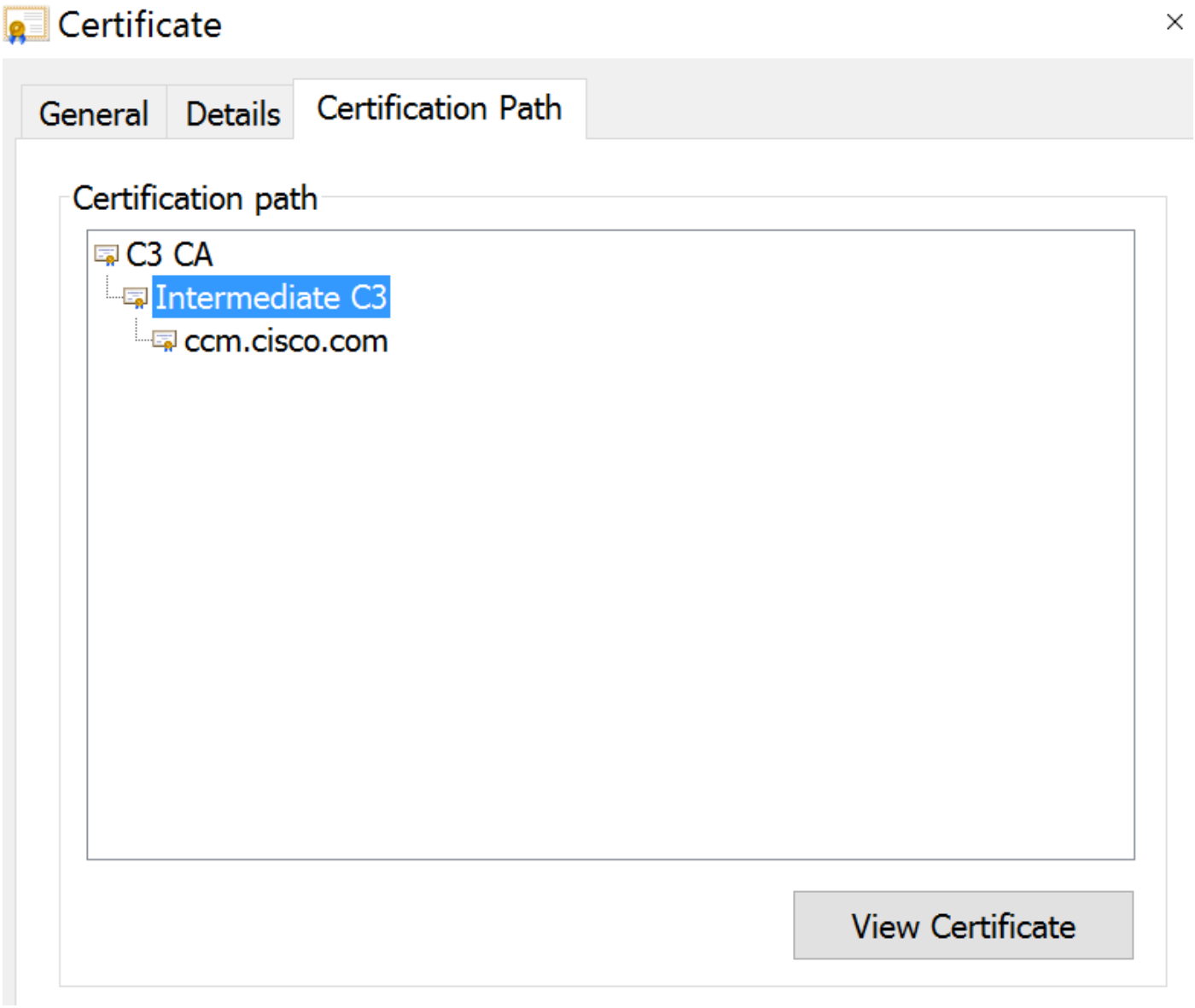
ステップ 25。 イメージに示すように『Details』 を選択 して下さい。



ステップ 26。 イメージに示すように**ファイルに**『Copy』 を選択 して下さい。



ステップ 27。 信頼できない CA についてのエラーを得る場合、中間物およびルート証明を表示するために**認証パス**にナビゲート して下さい。 またそれらをクリックし、認証を表示し、ファイルにイメージに示すようにそれらをコピーできます。



ステップ 28。 認証をダウンロードしてもらったら信頼された機関および中間機関としてこれらの認証をインストールするオペレーティング システム（OS）またはブラウザの手順に従って下さい。