

# MRAサービスのためのExpresswayトラフィックサーバ証明書検証のトラブルシューティング

## 内容

---

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[信頼済みCAチェーン](#)

[SANまたはCNの確認](#)

[動作の変更](#)

[X14.2.0より前のバージョン](#)

[X14.2.0以降のバージョン](#)

[シナリオのトラブルシューティング](#)

[1. リモート証明書に署名したCAが信頼されていない](#)

[2. 証明書に接続アドレス \( FQDNまたはIP \) が含まれていない](#)

[簡単に検証する方法](#)

[解決方法](#)

[関連情報](#)

---

## はじめに

このドキュメントでは、Cisco Bug ID [CSCwc69661](#)またはCisco Bug ID [CSCwa25108](#)に関連するExpresswayバージョンX14.2.0以降の動作の変更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。

- Expresswayの基本設定
- MRA基本設定

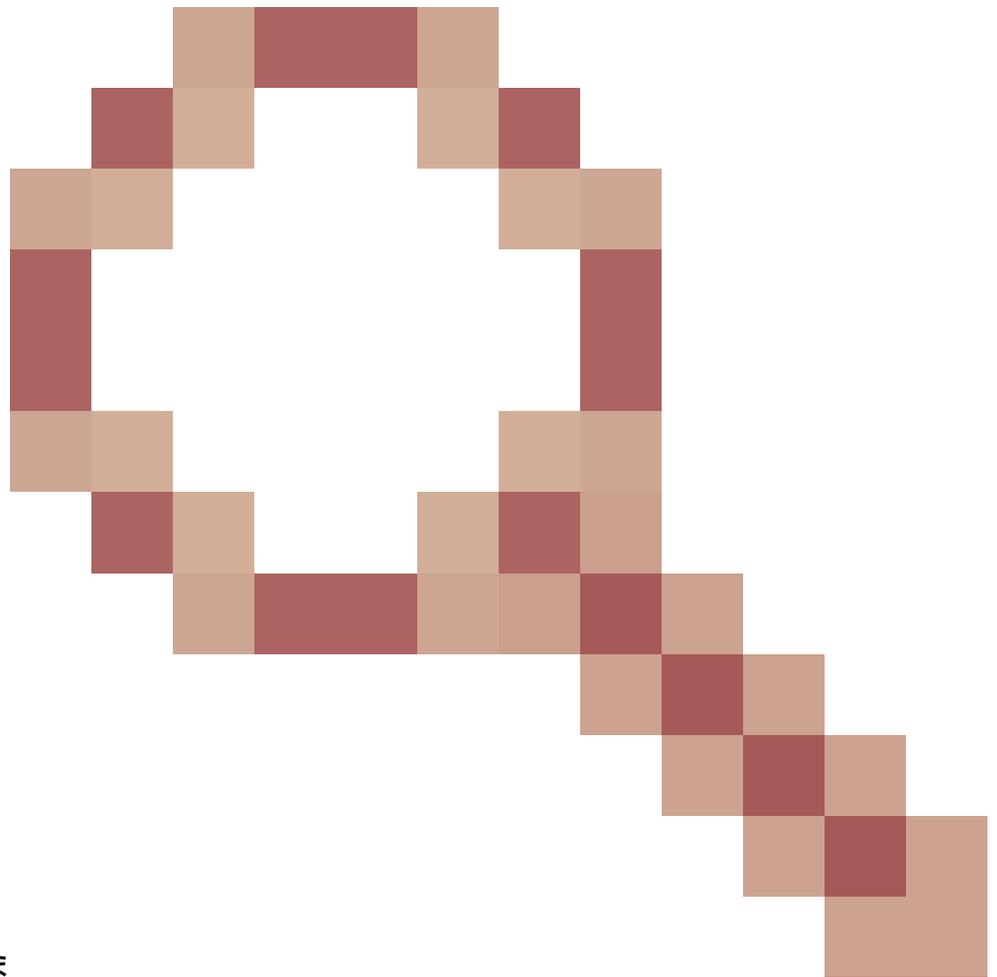
### 使用するコンポーネント

このドキュメントの情報は、Cisco ExpresswayバージョンX14.2以降に基づくものです。

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。

## 背景説明



Cisco Bug ID [CSCwc69661](#) まで

またはCisco Bug ID [CSCwa25108](#) で示されているこの動作変更により、Expresswayプラットフォーム(MRA)上のトラフィックサーバは、モバイルおよびリモートアクセス(MRA)サービス用にCisco Unified Communication Manager(CUCM)、Cisco Unified Instant Messaging & Presence(IM&P)、およびUnityサーバノードの証明書などの検証を実行します。この変更により、Expresswayプラットフォームでのアップグレード後にMRAログイン障害が発生する可能性があります。

Hypertext Transfer Protocol Secure(HTTPS)は、Transport Layer Security(TLS)を使用して通信を暗号化する安全な通信プロトコルです。このセキュアチャネルは、TLSハンドシェイクで交換されるTLS証明書を使用して作成されます。このサーバには、認証(リモート側の接続先を知る)とプライバシー(暗号化)の2つの目的があります。認証は中間者攻撃から保護し、プライバシーは攻撃者が通信を傍受して改ざんするのを防ぎます。

TLS(証明書)検証は認証の観点で実行され、正しいリモートの当事者に接続していることを確認できません。検証は、次の2つの個別の項目で構成されます。

1. 信頼された認証局(CA)チェーン
2. サブジェクトの別名(SAN)または共通名(CN)

## 信頼済みCAチェーン

Expressway-CがCUCM / IM&P / Unityが送信する証明書を信頼するには、信頼するトップレベル（ルート）認証局(CA)へのその証明書からのリンクを確立する必要があります。このようなリンク、つまりエンティティ証明書をルートCA証明書にリンクする証明書の階層は、信頼のチェーンと呼ばれます。このような信頼のチェーンを確認できるように、各証明書には、Issuer（または「Issued by」）とSubject（または「Issued To」）の2つのフィールドが含まれています。

CUCMがExpressway-Cに送信するサーバ証明書などのサーバ証明書は、「件名」フィールドに、通常はCN内の完全修飾ドメイン名(FQDN)を持っています。

```
Issuer: DC=lab, DC=vngtp, CN=vngtp-ACTIVE-DIR-CA
Subject: C=BE, ST=Flamish-Brabant, L=Diegem, O=Cisco, OU=TAC, CN=cucm.vngtp.lab
```

CUCM cucm.vngtp.labのサーバ証明書例SubjectフィールドのCN属性に、Country(C)、State(ST)、Location(L)、...などの他の属性とともにFQDNが含まれています。また、サーバ証明書がvngtp-ACTIVE-DIR-CAという名前のCAによって配布（発行）されることもわかります。

トップレベルCA（ルートCA）は、自身を識別する証明書を発行することもできます。このようなルートCA証明書では、IssuerとSubjectの値が同じであることがわかります。

```
Issuer: DC=lab, DC=vngtp, CN=vngtp-ACTIVE-DIR-CA
Subject: DC=lab, DC=vngtp, CN=vngtp-ACTIVE-DIR-CA
```

これは、ルートCAが自身を識別するために配布する証明書です。

一般的な状況では、ルートCAはサーバ証明書を直接発行しません。代わりに、他のCAの証明書を発行します。このような他のCAは、中間CAと呼ばれます。中間CAは、サーバ証明書または他の中間CA用の証明書を直接発行できます。サーバ証明書が中間CA 1によって発行され、次に中間CA 2から証明書が取得されるという状況が考えられます。最終的に中間CAがルートCAから自身の証明書を直接取得するまで（証明書はルートCAから直接取得されます）、

Server certificate :

```
Issuer: DC=lab, DC=vngtp, CN=vngtp-intermediate-CA-1
Subject: C=BE, ST=Flamish-Brabant, L=Diegem, O=Cisco, OU=TAC, CN=cucm.vngtp.lab
```

Intermediate CA 1 certificate :

```
Issuer: DC=lab, DC=vngtp, CN=vngtp-intermediate-CA-2
Subject: DC=lab, DC=vngtp, CN=vngtp-intermediate-CA-1
```

Intermediate CA 2 certificate :

```
Issuer: DC=lab, DC=vngtp, CN=vngtp-intermediate-CA-3
Subject: DC=lab, DC=vngtp, CN=vngtp-intermediate-CA-2
```

...

Intermediate CA n certificate :

```
Issuer: DC=lab, DC=vngtp, CN=vngtp-ACTIVE-DIR-CA
Subject: DC=lab, DC=vngtp, CN=vngtp-intermediate-CA-n
```

Root CA certificate :

Issuer: DC=lab, DC=vngtp, CN=vngtp-ACTIVE-DIR-CA

Subject: DC=lab, DC=vngtp, CN=vngtp-ACTIVE-DIR-C

ここで、Expressway-CがCUCMから送信されるサーバ証明書を信頼するためには、そのサーバ証明書からルートCA証明書までの信頼チェーンを構築する必要があります。そのためには、ルートCA証明書およびすべての中間CA証明書を ( 存在する場合は、ルートCAがCUCMのサーバ証明書を直接発行した場合を除いて ) Expressway-Cの信頼ストアにアップロードする必要があります。

 注：IssuerフィールドとSubjectフィールドは、信頼のチェーンを人間が読める方法で簡単に構築できますが、CUCMでは、これらのフィールドを証明書で使用しません。代わりに、「X509v3 Authority Key Identifier」フィールドと「X509v3 Subject Key Identifier」フィールドを使用して、信頼のチェーンを構築します。これらのキーには、Subject/Issuerフィールドを使用するよりも正確な証明書のIDが含まれています。同じSubject/Issuerフィールドを持つ2つの証明書が存在する可能性があります、そのうちの1つは期限切れであり、1つはまだ有効です。両方とも異なるX509v3 Subject Key Identifier(SUBJECT KEY ID)を持つため、CUCMは正しい信頼チェーンを引き続き判別できます。

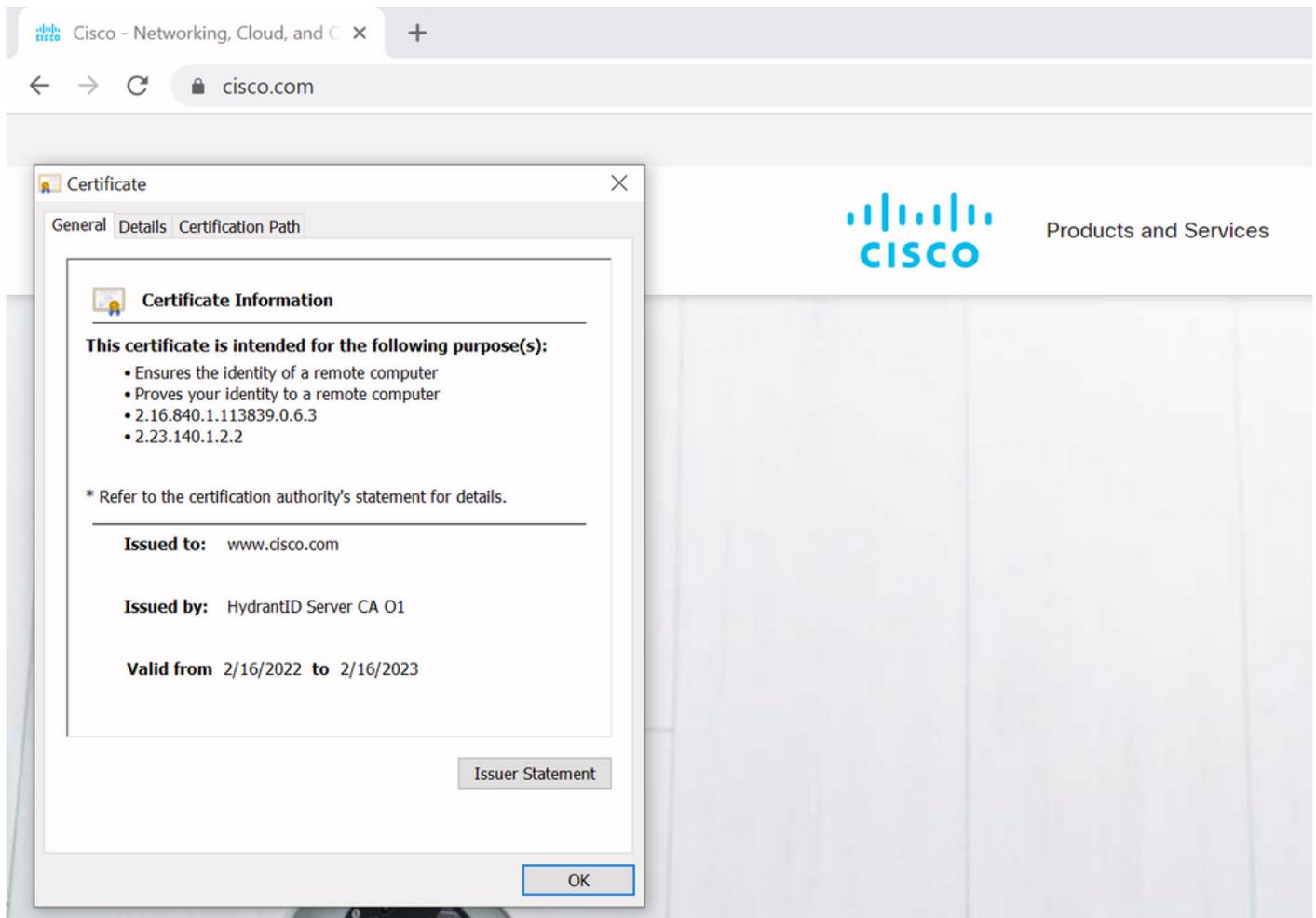
Cisco Bug ID [CSCwa12905](#)によると、これはExpresswayには該当せず、同じ共通名(CN)を持つ2つの異なる ( 自己署名されたなど ) 証明書をExpresswayの信頼ストアにアップロードすることはできません。これを修正する方法は、CA署名付き証明書を使用するか、異なる共通名(CN)を使用するか、または常に同じ証明書を使用していることを確認する方法です ( CUCM 14の再使用証明書機能を使用する可能性があります ) 。

## SANまたはCNの確認

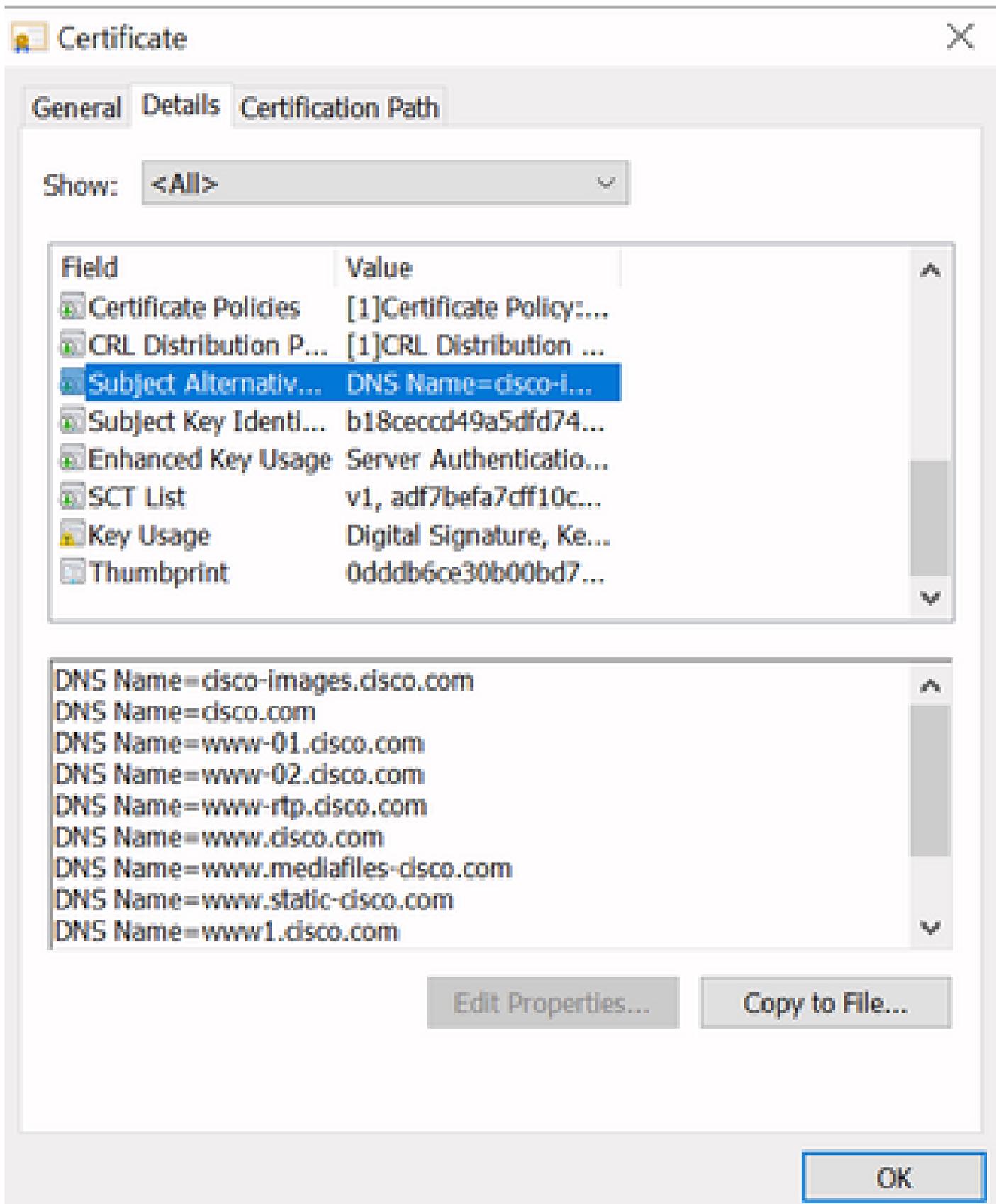
ステップ1は信頼ストアをチェックアウトしますが、信頼ストア内のCAによって署名された証明書を持つユーザは、その時点で有効になります。これは明らかに十分ではありません。したがって、特に接続するサーバが正しいものであるかどうかを検証する追加チェックがあります。これは、要求が行われたアドレスに基づいて行われます。

ブラウザでも同様の操作が行われますので、例を挙げて見てみましょう。

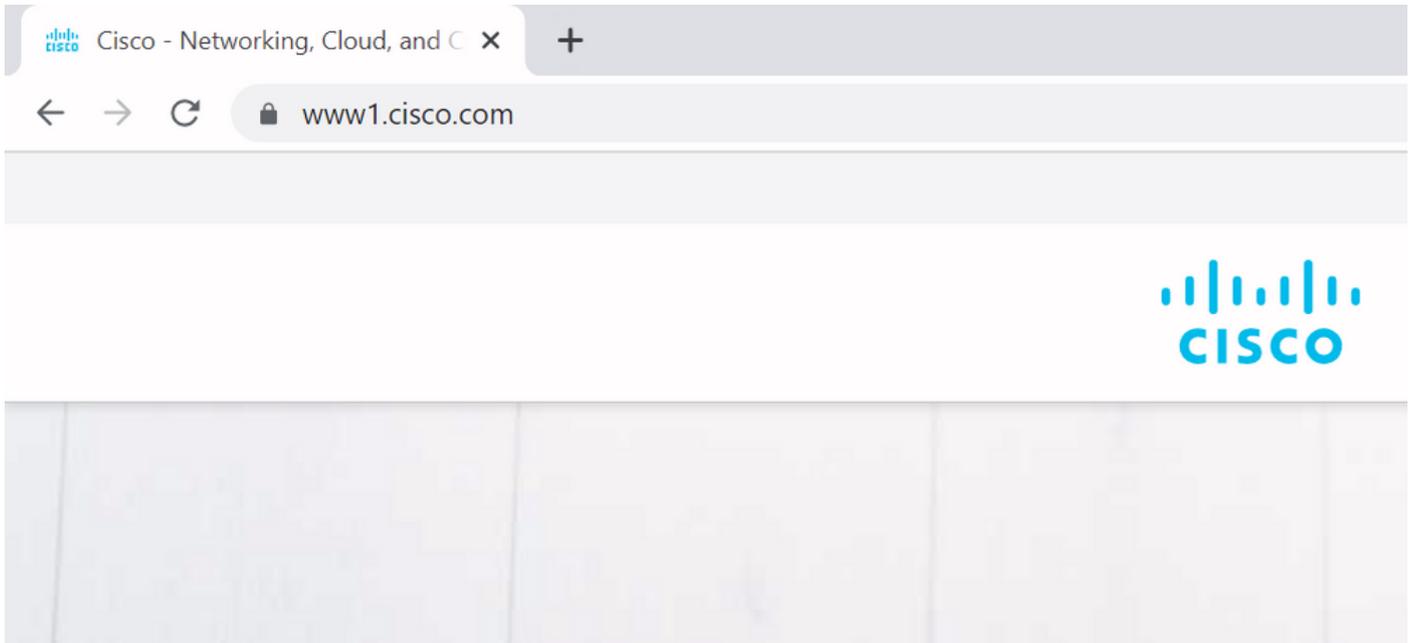
<https://www.cisco.com>を参照すると、入力したURLの横にロックアイコンが表示され、それが信頼できる接続であることを意味します。これは、CA信頼チェーン ( 最初のセクションから ) と SANまたはCNチェックの両方に基づいています。証明書を ( ブラウザのロックアイコンをクリックして ) 開くと、共通名 ( 「Issued to:」フィールドに表示 ) が[www.cisco.com](https://www.cisco.com)に設定されており、接続先のアドレスに正確に対応していることがわかります。この方法で、正しいサーバに確実に接続できます ( 証明書に署名し、証明書を配布する前に検証を実行するCAを信頼するため ) 。



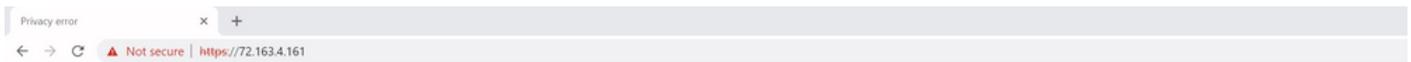
証明書の詳細、特にSANエントリを調べると、同じことが他のFQDNと同様に繰り返されていることがわかります。



つまり、たとえば<https://www1.cisco.com>への接続を要求する場合、SANエントリに含まれているため、セキュアな接続としても表示されます。



ただし、<https://www.cisco.com>をブラウズせず、IPアドレス(<https://72.163.4.161>)を直接参照する場合、署名したCAを信頼しますが、提示された証明書にはサーバへの接続に使用したアドレス(72.163.4.161)が含まれていないため、セキュアな接続は表示されません。



```
Command Prompt - nslookup
C:\Users\stejanss>
C:\Users\stejanss>nslookup
Default Server: dns-aer1.cisco.com
Address: 173.38.200.100

> cisco.com
Server: dns-aer1.cisco.com
Address: 173.38.200.100

Name: cisco.com
Addresses: 2001:420:1101:1::a
           72.163.4.161
>
```

**Your connection is not private**

Attackers might be trying to steal your information from **72.163.4.161** (for example, passwords, messages, or credit cards). [Learn more](#)

NET:ERR\_CERT\_COMMON\_NAME\_INVALID

To get Chrome's highest level of security, [turn on enhanced protection](#)

This server could not prove that it is **72.163.4.161**; its security certificate is from **www.cisco.com**. This may be caused by a misconfiguration or an attacker intercepting your connection.

[Proceed to 72.163.4.161 \(unsafe\)](#)

ブラウザでは、これをバイパスできますが、これはバイパスが許可されていないTLS接続で有効にできる設定です。そのため、証明書には、リモートのユーザが接続に使用する予定の正しいCNまたはSAN名が含まれていることが重要です。

## 動作の変更

MRAサービスは、CUCM/IM&P/Unityサーバに対するExpressway経由の複数のHTTPS接続に大きく依存して、適切に認証し、ログインするクライアントに固有の適切な情報を収集します。この

通信は通常、ポート8443および6972経由で行われます。

## X14.2.0より前のバージョン

X14.2.0より前のバージョンでは、これらのセキュアなHTTPS接続を処理するExpressway-Cのトラフィックサーバは、リモートエンドから提示された証明書を確認しませんでした。これにより、中間者攻撃が発生する可能性があります。MRA設定には、Configuration > Unified Communications > Unified CM servers / IM and Presence Service nodes / Unity Connection serversでCUCM / IM&P / Unityサーバを追加するときに、「TLS Verify Mode」を「On」に設定してTLS証明書を検証するオプションがあります。例として、設定オプションと関連情報ボックスを示します。この例では、SAN内のFQDNまたはIP、証明書の有効性、および証明書が信頼できるCAによって署名されているかどうかを確認します。



Cisco Expressway-C

Status > System > Configuration > Applications > Users > Maintenance >

### Unified CM servers You are here: [Configuration](#)

#### Unified CM server lookup

Unified CM publisher address	cucmpub.vngtp.lab
Username	* administrator <span>i</span>
Password	* ..... <span>i</span>
<b>TLS verify mode</b>	On <span>i</span>
Deployment	Default deployment <span>i</span>
AES GCM support	Off <span>i</span>
SIP UPDATE for session refresh	Off <span>i</span>
ICE Passthrough support	Off <span>i</span>

このTLS証明書検証チェックは、CUCM/IM&P/Unityサーバの検出時にのみ実行され、MRAログイン中にさまざまなサーバが照会されることはありません。この設定の最初の欠点は、追加したパブリッシャアドレスに対してのみ確認が行われることです。パブリッシャノードのデータベースからサブスクリバノード情報 ( FQDNまたはIP ) を取得するため、サブスクリバノード上の証明書が正しく設定されているかどうかは検証されません。この設定の2つ目の欠点は、接続情報がExpressway-C設定に配置されたパブリッシャアドレスと異なる場合があるため、MRAクライアントにアドバタイズされる情報が異なる点です。たとえばCUCMのSystem > Serverで、サーバをIPアドレス ( たとえば10.48.36.215 ) でアドバタイズし、それを ( プロキシされたExpressway接続を介して ) MRAクライアントが使用します。ただし、FQDNをcucm.steven.labにしてExpressway-CのCUCMに追加できます。したがって、CUCMのtomcat証

明書にIPアドレスではなくSANエントリとしてcucm.steven.labが含まれている場合、「TLS検証モード」を「オン」に設定した検出は成功しますが、MRAクライアントからの実際の通信は異なるFQDNまたはIPを対象とすることができるため、TLS検証は失敗します。

## X14.2.0以降のバージョン

X14.2.0バージョン以降では、Expresswayサーバは、トラフィックサーバを介して行われる個々のHTTPS要求に対して、TLS証明書の検証を実行します。つまり、CUCM/IM&P/Unityノードの検出中に「TLS Verify Mode」が「Off」に設定されている場合にも、これが実行されます。検証が成功しない場合、TLSハンドシェイクが完了せず、要求が失敗します。その結果、冗長性やフェールオーバーの問題などの機能が失われたり、ログイン障害が完了したりする可能性があります。また、「TLS検証モード」を「オン」に設定しても、すべての接続が後の例で説明するように正常に機能するとは限りません。

ExpresswayがCUCM / IM&P / Unityノードに対してチェックする正確な証明書は、『[MRAガイド](#)』のセクションに示すとおりです。

TLS検証のデフォルトの他に、X14.2で導入された変更もあります。これにより、アップグレードパスに応じて、暗号リストの異なる優先順位がアダプタイズされる可能性があります。アップグレード前にCUCM（またはECDSAアルゴリズム用の個別の証明書を持つその他の製品）からCisco TomcatまたはCisco CallManager証明書を要求する一方で、アップグレード後にはECDSAバリエーション（実際にはRSAよりも安全な暗号バリエーション）を要求する必要があるため、ソフトウェアアップグレード後に予期しないTLS接続が発生する可能性があります。Cisco Tomcat-ECDSAまたはCisco CallManager-ECDSA証明書は、別のCAによって署名することも、自己署名証明書だけで署名することもできます（デフォルト）。

この暗号の優先順位の変更は、Expressway X14.2.1の[リリースノート](#)で示されているようにアップグレードパスによって異なるため、常に関連するわけではありません。要するに、「ECDHE-RSA-AES256-GCM-SHA384:」をプリペンドするかどうか、各暗号リストのMaintenance > Security > Ciphersで確認できます。そうでない場合は、RSA暗号よりも新しいECDSA暗号が優先されます。存在する場合は、以前のRSAの動作よりも優先度が高い動作になります。

### Cipher Preferences - ECDSA Cipher Preference Over RSA

ECDSA certificates are preferred over RSA.



#### Important

The following points lists the various upgrade path(s) that are mandatory for upgrading ciphers.

1. When upgrading from version lower than 14.0 to 14.2, the ECDSA would be preferred. If you prefer RSA certificates over ECDSA, then prefix the cipher string with "ECDHE-RSA-AES256-GCM-SHA384:" using either Web User Interface (Maintenance > Security > Ciphers) or CLI command (xConfiguration Ciphers).
2. When upgrading from version equal or higher than 14.0 to 14.2 or higher version, you have appended "ECDHE-RSA-AES256-GCM-SHA384:" to the default Ciphers List to prefer RSA certificates over ECDSA. If you prefer ECDSA certificates over RSA, then remove "ECDHE-RSA-AES256-GCM-SHA384:" from the cipher string using Web User Interface (Maintenance > Security > Ciphers) or CLI command (xConfiguration Ciphers).
3. Any customer has a fresh install X14.2 image, ECDSA is being preferred. If you prefer RSA certificates over ECDSA, then prefix the cipher string with "ECDHE-RSA-AES256-GCM-SHA384:" using either Web User Interface (Maintenance > Security > Ciphers) or CLI command (xConfiguration Ciphers).

このシナリオでTLS検証が失敗する可能性のある方法は2つあります。これについては後で詳しく説明します。

1. リモート証明書に署名したCAが信頼されていない

a. 自己署名証明書

b. 不明なCAによって署名された証明書

2. 証明書に接続アドレス（FQDNまたはIP）が含まれていない

# シナリオのトラブルシューティング

次のシナリオは、ExpresswayをX14.0.7からX14.2にアップグレードした後にMRAログインが失敗したラボ環境でも同様のシナリオを示しています。ログは類似していますが、解決方法が異なります。ログは、MRAログイン前に開始し、MRAログイン失敗後に停止した診断ログ (Maintenance > Diagnostics > Diagnostic loggingから)によって収集されます。追加のデバッグロギングは有効にされていません。

## 1. リモート証明書に署名したCAが信頼されていない

リモート証明書は、Expressway-Cの信頼ストアに含まれていないCAによって署名されているか、Expressway-Cサーバの信頼ストアに追加されていない自己署名証明書 (本質的にはCA) である可能性があります。

次の例では、CUCMに送信される要求(10.48.36.215 - cucm.steven.lab)がポート8443 ( 200 OK応答 ) で正しく処理されますが、TFTP接続に対してポート6972でエラー ( 502応答 ) がスローされることを確認できます。

```
<#root>
```

```
===Success connection on 8443===
```

```
2022-07-11T18:55:25.910+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,910" Module="net
```

```
2022-07-11T18:55:25.917+02:00 vcsc traffic_server[18242]: Event="Request Allowed" Detail="Access allow
```

```
2022-07-11T18:55:25.917+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,916" Module="net
```

```
2022-07-11T18:55:25.955+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,955" Module="net
```

```
2022-07-11T18:55:25.956+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,955" Module="net
```

```
200
```

```
"
```

```
===Failed connection on 6972===
```

```
2022-07-11T18:55:26.000+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:26,000" Module="net
```

```
2022-07-11T18:55:26.006+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:26,006" Module="net
```

```
2022-07-11T18:55:26.016+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:26,016" Module="net
```

```
2022-07-11T18:55:26.016+02:00 vcsc traffic_server[18242]: [ET_NET 0]
```

```
WARNING: Core server certificate verification failed for
```

```
(cucm.steven.lab).
```

```
Action=Terminate Error=self signed certificate server=cucm.steven.lab(10.48.36.215)
```

```
depth=0
```

```
2022-07-11T18:55:26.016+02:00 vcsc traffic_server[18242]: [ET_NET 0]
```

```
ERROR: SSL connection failed for
```

```
'cucm.steven.lab': error:1416F086:
```

SSL routines:tls\_process\_server\_certificate:certificate verify failed

2022-07-11T18:55:26.024+02:00 vcsd traffic\_server[18242]: UTCTime="2022-07-11 16:55:26,024" Module="net

502 connect failed

"

「certificate verify failed」というエラーは、Expressway-CがTLSハンドシェイクを検証できなかったことを示しています。その理由は、自己署名証明書を示すため、警告行に表示されます。深さが0と表示されている場合は、自己署名証明書です。深さが0よりも大きい場合は、証明書チェーンがあることを意味し、不明なCAによって署名されています ( Expressway-Cから見た場合 )。

テキストログに示されているタイムスタンプで収集されたpcapファイルを見ると、CUCMがCNがcucm-ms.steven.lab ( およびSANがcucm.steven.lab ) で、steven-DC-CAによって署名された証明書を、ポート8443でExpressway-Cに提示していることがわかります。

The screenshot shows a Wireshark capture of a TLS handshake. The 'Details' pane for the failed certificate verification shows the following structure:

- Certificate (2423 bytes)
  - Certificate Length: 1507
  - Signature: 30E205F20E204C7A0030201020134500001220956058D3... (id-at-commonName=cucm-ms.steven.lab, id-at-organizationalUnitName=TAC, id-at-organizationName=Cisco, id-at-localityName=Oiegen, id-at-stateOrProvinceName=Belgium, id-at-countryName=BE)
    - version: v3 (2)
    - serialNumber: 0x45000001220956058D34860844200200000122
    - signature (sha1withRSAEncryption)
    - issuer: rdnsSequence (0)
    - validity
    - subject: rdnsSequence (0)
    - subjectPublicKeyInfo
    - extensions: 9 items
      - Extension (id-ce-keyusage)
      - Extension (id-ce-keyusage)
      - Extension (id-ce-subjectAltName)
        - Extension Id: 2.5.29.17 (id-ce-subjectAltName)
          - critical: True
          - GeneralNames: 3 items
            - GeneralName: dNSName (2)
              - dNSName: cups.steven.lab
              - dNSName: steven.lab
              - dNSName: cucm.steven.lab
            - Extension (id-ce-subjectKeyIdentifier)
            - Extension (id-ce-authorityKeyIdentifier)
            - Extension (id-ce-cRLDistributionPoints)
            - Extension (id-pe-authorityInfoAccessSyntax)
            - Extension (id-ms-certificate-template)
            - Extension (id-ms-application-certificate-policies)
            - algorithmIdentifier (sha1withRSAEncryption)
            - padding: 0
              - encrypted: 9fb07874637a2a92071ef608f22709cecc7ec44470c82b...

しかし、ポート6972で提示された証明書を調べると、これがcucm-EC.steven.labとして設定されたCNを持つ自己署名証明書 ( 発行者自身 ) であることがわかります。-EC拡張子は、これがCUCMで設定されているECDSA証明書であることを示します。

eth0\_diagnostic\_logging\_tcpdump00\_vsc\_2022-07-11\_16\_55\_44.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcpdump4972

No.	Time	Source	Src port	Destination	Dest port	Protocol	OSFP	VLAs	Length	Info
4736	2022-07-11 16:55:26.006608	10.48.36.46		15176 10.48.36.215	6972 TCP	C50		74	31576 > 6972 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=478578525 TSecr=0 WS=128	
4737	2022-07-11 16:55:26.006852	10.48.36.215		6972 10.48.36.46	31576 TCP	C50		74	6972 > 31576 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=343633320 TSecr=478578525 WS=128	
4732	2022-07-11 16:55:26.007300	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		66	31576 > 6972 [ACK] Seq=1 Win=64256 Len=0 TSval=478578525 TSecr=343633320	
4733	2022-07-11 16:55:26.007300	10.48.36.46		6972 15176.2	6972 TLSv1.2	C50		583	Client Hello	
4734	2022-07-11 16:55:26.013050	10.48.36.215		6972 10.48.36.46	31576 TLSv1.2	C50		1514	Server Hello, Certificate, Server Key Exchange	
4735	2022-07-11 16:55:26.013911	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		66	31576 > 6972 [ACK] Seq=518 Ack=1449 Win=64128 Len=0 TSval=478578535 TSecr=343633329	
4736	2022-07-11 16:55:26.016400	10.48.36.215		6972 10.48.36.46	31576 TLSv1.2	C50		499	Certificate Request, Server Hello Done	
4737	2022-07-11 16:55:26.016419	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		66	31576 > 6972 [ACK] Seq=518 Ack=1882 Win=63744 Len=0 TSval=478578535 TSecr=343633329	
4738	2022-07-11 16:55:26.016703	10.48.36.46		31576 10.48.36.215	6972 TLSv1.2	C50		73	Alert (Level: Fatal, Description: Unknown CA)	
4739	2022-07-11 16:55:26.016821	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		74	31576 > 6972 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=478578535 TSecr=0 WS=128	
4740	2022-07-11 16:55:26.016965	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		66	31576 > 6972 [RST, ACK] Seq=525 Ack=1882 Win=64128 Len=0 TSval=478578535 TSecr=343633329	
4741	2022-07-11 16:55:26.016984	10.48.36.215		6972 10.48.36.46	31576 TCP	C50		74	6972 > 31576 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=343633330 TSecr=478578535 WS=128	
4742	2022-07-11 16:55:26.017009	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		66	31576 > 6972 [ACK] Seq=1 Win=64256 Len=0 TSval=478578535 TSecr=343633330	
4743	2022-07-11 16:55:26.017181	10.48.36.215		6972 10.48.36.46	31576 TCP	C50		66	6972 > 31576 [SYN, ACK] Seq=1882 Ack=54 Win=30688 Len=0 TSval=343633330 TSecr=478578535	
4744	2022-07-11 16:55:26.017212	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		54	31576 > 6972 [RST] Seq=525 Win=0 Len=0	
4745	2022-07-11 16:55:26.017218	10.48.36.46		31576 10.48.36.215	6972 TLSv1.2	C50		583	Client Hello	
4746	2022-07-11 16:55:26.024226	10.48.36.215		6972 10.48.36.46	31576 TLSv1.2	C50		1514	Server Hello, Certificate, Server Key Exchange	
4747	2022-07-11 16:55:26.024265	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		66	31576 > 6972 [ACK] Seq=518 Ack=1449 Win=64128 Len=0 TSval=478578543 TSecr=343633337	
4748	2022-07-11 16:55:26.024298	10.48.36.215		31576 10.48.36.46	31576 TLSv1.2	C50		590	Certificate Request, Server Hello Done	
4749	2022-07-11 16:55:26.024309	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		66	31576 > 6972 [ACK] Seq=518 Ack=1883 Win=63744 Len=0 TSval=478578543 TSecr=343633337	
4750	2022-07-11 16:55:26.024548	10.48.36.46		31576 10.48.36.215	6972 TLSv1.2	C50		73	Alert (Level: Fatal, Description: Unknown CA)	
4751	2022-07-11 16:55:26.024647	10.48.36.46		31576 10.48.36.215	6972 TCP	C50		66	31576 > 6972 [RST, ACK] Seq=525 Ack=1883 Win=64128 Len=0 TSval=478578543 TSecr=343633337	
4767	2022-07-11 16:55:26.033159	10.48.36.46		31500 10.48.36.215	6972 TCP	C50		74	31500 > 6972 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=478578601 TSecr=0 WS=128	

```

Secure Sockets Layer
  TLSv1.2 Record Layer: Handshake Protocol: Server Hello
  TLSv1.2 Record Layer: Handshake Protocol: Certificate
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 667
  Handshake Protocol: Certificate
    Handshake type: Certificate (11)
    Length: 663
    Certificates length: 660
    Certificates (600 Bytes)
      Certificate Length: 657
      Certificate: 30820282082021480830202107470ee6271e1d346... (id-at-localityName=diegem,id-at-stateOrProvinceName=Belgium,id-at-commonName=cucm-ec.steven.lab,id-at-organizationalUnitName=TAC,id-at-organizationName=Cisco,id-at-countryName=)
        version: v3 (2)
        serialNumber: 02470ee6271e1d3461099460a30f5fd
        signature (ecdsa-with-SHA384)
        issuer: rdmsncertificate (8)
        consequences: 6 items (id-at-localityName=diegem,id-at-stateOrProvinceName=Belgium,id-at-commonName=cucm-ec.steven.lab,id-at-organizationalUnitName=TAC,id-at-organizationName=Cisco,id-at-countryName=EE)
        validity
        subject: rdmsncertificate (8)
        subjectPublicKeyInfo
        extensions: 5 items
          Extension (id-ce-keyUsage)
          Extension (id-ce-extendedKeyUsage)
          Extension (id-ce-subjectKeyIdentifier)
          Extension (id-ce-basicConstraints)
          Extension (id-ce-subjectAltName)
            Extension Id: 2.5.29.17 (id-ce-subjectAltName)
              GeneralNames: 1 item
                GeneralName (2)
                  dnName: cucm.steven.lab
                  algorithmIdentifier (ecdsa-with-SHA384)
                  padding: 0
                  encrypted: 306402021480830202107470ee6271e1d346...
  TLSv1.2 Record Layer: Handshake Protocol: Server Key Exchange

```

CUCMのCisco Unified OS Administrationで、次のようにSecurity > Certificate Managementの下に配置された証明書を確認できます。tomcatとtomcat-ECDSA用の異なる証明書が表示されます。この証明書では、tomcatはCA署名付き ( Expressway-Cによって信頼されます ) ですが、tomcat-ECDSA証明書は自己署名付きで、Expressway-Cによって信頼されません。

Cisco Unified Operating System Administration

Navigation: Cisco Unified OS Administration

Security > Software Updates > Services > Help

Generate Self-signed Upload Certificate/Certificate chain Download CTL Generate CSR Download CSR

Status

43 records found

Certificate	Common Name	Type	Key Type	Distribution	Issued By	Expiration	Description
auth2	AUTH2_cucm.steven.lab	Self-signed	RSA	cucm.steven.lab	AUTH2_cucm.steven.lab	07/21/2038	Self-signed certificate generated by system
CallManager	cucm.steven.lab	CA-signed	EC	cucm.steven.lab	stevan-DC-CA	07/13/2022	Certificate Signed by stevan-DC-CA
CallManager-ECDSA	cucm-ec.steven.lab	Self-signed	EC	cucm.steven.lab	cucm-ec.steven.lab	02/18/2024	Self-signed certificate generated by system
CallManager-trust	stevan-DC-CA	Self-signed	EC	stevan-DC-CA	stevan-DC-CA	06/01/2023	Signed Certificate
CallManager-trust	NOMAT-CA-CA	Self-signed	EC	NOMAT-CA-CA	NOMAT-CA-CA	04/23/2028	Signed Certificate
CallManager-trust	CAP-RTP-002	Self-signed	EC	CAP-RTP-002	CAP-RTP-002	10/10/2023	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CallManager-trust	CAPF-eb206488	Self-signed	EC	CAPF-eb206488	CAPF-eb206488	04/12/2020	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CallManager-trust	ms-AD2-CA-1	Self-signed	EC	ms-AD2-CA-1	ms-AD2-CA-1	09/11/2024	vngtp-CA
CallManager-trust	CAP-RTP-001	Self-signed	EC	CAP-RTP-001	CAP-RTP-001	02/07/2023	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CallManager-trust	NOMAT-CA-10	Self-signed	EC	NOMAT-CA-10	NOMAT-CA-10	08/11/2027	Signed Certificate
CallManager-trust	Cisco_Root_CA_H2	Self-signed	EC	Cisco_Root_CA_H2	Cisco_Root_CA_H2	11/12/2037	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CallManager-trust	ACT2_SU02_CA	CA-signed	EC	ACT2_SU02_CA	Cisco_Root_CA_2048	05/14/2029	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CallManager-trust	vngtp-ACTIVE-ORR-CA	Self-signed	EC	vngtp-ACTIVE-ORR-CA	Cisco_Root_CA_H2	02/10/2024	Self-signed certificate generated by system
CallManager-trust	Cisco_Root_CA_2048	Self-signed	EC	Cisco_Root_CA_2048	Cisco_Root_CA_2048	05/14/2029	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CallManager-trust	Cisco_Manufacturing_CA_SHA2	CA-signed	EC	Cisco_Manufacturing_CA_SHA2	Cisco_Root_CA_H2	11/12/2037	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CallManager-trust	dcomics-WONDERWOMAN-CA	Self-signed	EC	dcomics-WONDERWOMAN-CA	Cisco_Manufacturing_CA_SHA2	09/19/2037	CA-bantrum
CallManager-trust	CAPF-6164213c	Self-signed	EC	CAPF-6164213c	CAPF-6164213c	07/12/2025	Self-signed certificate generated by system
CAPF	CAPF-6164213c	Self-signed	EC	CAPF-6164213c	CAPF-6164213c	07/12/2025	Self-signed certificate generated by system
CAPF-trust	CAP-RTP-002	Self-signed	EC	CAP-RTP-002	CAP-RTP-002	10/10/2023	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CAPF-trust	CAPF-eb206488	Self-signed	EC	CAPF-eb206488	CAPF-eb206488	04/12/2020	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CAPF-trust	CAP-RTP-001	Self-signed	EC	CAP-RTP-001	CAP-RTP-001	02/07/2023	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CAPF-trust	Cisco_Root_CA_H2	Self-signed	EC	Cisco_Root_CA_H2	Cisco_Root_CA_H2	11/12/2037	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CAPF-trust	ACT2_SU02_CA	CA-signed	EC	ACT2_SU02_CA	Cisco_Root_CA_2048	05/14/2029	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CAPF-trust	Cisco_Root_CA_2048	Self-signed	EC	Cisco_Root_CA_2048	Cisco_Root_CA_2048	05/14/2029	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CAPF-trust	Cisco_Manufacturing_CA_SHA2	CA-signed	EC	Cisco_Manufacturing_CA_SHA2	Cisco_Root_CA_H2	11/12/2037	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CAPF-trust	Cisco_Manufacturing_CA_SHA2	CA-signed	EC	Cisco_Manufacturing_CA_SHA2	Cisco_Root_CA_H2	11/12/2037	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CAPF-trust	Cisco_Manufacturing_CA_SHA2	CA-signed	EC	Cisco_Manufacturing_CA_SHA2	Cisco_Root_CA_H2	11/12/2037	This certificate was used to sign the MIC installed on Cisco endpoint. Presence of this certificate allows the end point to communicate securely with UCH using the MIC when associated with a secure profile.
CAPF-trust	CAPF-6164213c	Self-signed	EC	CAPF-6164213c	CAPF-6164213c	07/12/2025	Self-signed certificate generated by system
ipsec	cucm.steven.lab	Self-signed	EC	cucm.steven.lab	cucm.steven.lab	07/12/2025	Trust Certificate
ipsec-trust	cucm.steven.lab	Self-signed	EC	cucm.steven.lab	cucm.steven.lab	07/12/2025	Trust Certificate
TLRECOVERY	TLRECOVERY_cucm.steven.lab	Self-signed	EC	cucm.steven.lab	TLRECOVERY_cucm.steven.lab	02/14/2039	Self-signed certificate generated by system
tomcat	cucm.steven.lab	Self-signed	EC	cucm.steven.lab	stevan-DC-CA	07/10/2024	Certificate Signed by stevan-DC-CA
tomcat-ECDSA	cucm-ec.steven.lab	Self-signed	EC	cucm.steven.lab	cucm-ec.steven.lab	07/25/2023	Self-signed certificate generated by system
tomcat-trust	stevan-DC-CA	Self-signed	EC	stevan-DC-CA	stevan-DC-CA	06/01/2023	Trust Certificate
tomcat-trust	NOMAT-CA-CA	Self-signed	EC	NOMAT-CA-CA	NOMAT-CA-CA	04/23/2028	Trust Certificate
tomcat-trust	cucm.steven.lab	Self-signed	EC	cucm.steven.lab	cucm.steven.lab	07/25/2023	Trust Certificate
tomcat-trust	cucm.steven.lab	Self-signed	EC	cucm.steven.lab	stevan-DC-CA	07/10/2024	Trust Certificate
tomcat-trust	cucm.steven.lab	Self-signed	EC	cucm.steven.lab	cucm.steven.lab	07/25/2023	Trust Certificate
tomcat-trust	NOMAT-CA-10	Self-signed	EC	NOMAT-CA-10	NOMAT-CA-10	08/11/2027	Signed Certificate
tomcat-trust	vngtp-ACTIVE-ORR-CA	Self-signed	EC	vngtp-ACTIVE-ORR-CA	vngtp-ACTIVE-ORR-CA	02/10/2024	Trust Certificate
tomcat-trust	dcomics-WONDERWOMAN-CA	Self-signed	EC	dcomics-WONDERWOMAN-CA	CA-bantrum	09/19/2037	Trust Certificate
TVS	cucm.steven.lab	Self-signed	EC	cucm.steven.lab	cucm.steven.lab	07/12/2025	Self-signed certificate generated by system

Generate Self-signed Upload Certificate/Certificate chain Download CTL Generate CSR Download CSR

## 2. 証明書に接続アドレス ( FQDNまたはIP ) が含まれていない

トラストストアとは別に、トラフィックサーバはMRAクライアントが要求を行う先の接続アドレ

スも確認します。たとえば、CUCMでIPアドレス(10.48.36.215)を使用してSystem > Serverを設定すると、Expressway-Cはそのようにクライアントにアドバタイズし、クライアントからの後続の要求 ( Expressway-C経由でプロキシされる ) はこのアドレスに向けて送信されます。

その特定の接続アドレスがサーバ証明書に含まれていない場合、TLS検証も失敗し、502エラーがスローされます。このエラーは、たとえばMRAログインが失敗する原因となります。

<#root>

```
2022-07-11T19:49:01.472+02:00 vcsc traffic_server[3916]: UTCTime="2022-07-11 17:49:01,472" Module="network" HTTPMSG:
```

```
|GET http://vcs_control.steven.lab:8443/c3RldmVuLmxhYi9odHRwcy8xMC40OC4zNi4yMTUvODQ0Mw/cucm-uds/user/emusk/...
...
```

```
2022-07-11T19:49:01.478+02:00 vcsc traffic_server[3916]: UTCTime="2022-07-11 17:49:01,478" Module="network" HTTPMSG:
```

```
2022-07-11T19:49:01.478+02:00 vcsc traffic_server[3916]: UTCTime="2022-07-11 17:49:01,478" Module="network" HTTPMSG:
```

```
|GET /cucm-uds/user/emusk/devices?max=100 HTTP/1.1
...
```

```
2022-07-11T19:49:01.491+02:00 vcsc traffic_server[3916]: [ET_NET 2]
```

```
WARNING: SNI (
```

```
10.48.36.215
```

```
) not in certificate
```

```
. Action=Terminate server=10.48.36.215(10.48.36.215)
```

```
2022-07-11T19:49:01.491+02:00 vcsc traffic_server[3916]: [ET_NET 2]
```

```
ERROR: SSL connection failed for
```

```
'10.48.36.215': error:1416F086:
```

```
SSL routines:tls_process_server_certificate:certificate verify failed
```

c3RldmVuLmxhYi9odHRwcy8xMC40OC4zNi4yMTUvODQ0Mwは(base64)を steven.lab/https/10.48.36.215/8443に変換します。これは、接続アドレスとして cucm.steven.labではなく10.48.36.215に接続する必要があることを示しています。パケットキャプチャに示すように、CUCM tomcat証明書にはSANのIPアドレスが含まれていないため、エラーがスローされます。

## 簡単に検証する方法

次の手順で、この動作が簡単に変更されるかどうかを検証できます。

1. Maintenance > Diagnostics > Diagnostic Loggingの順に選択し、Expressway-EおよびCサーバ (理想的にはTCPDumpsが有効な状態) の診断ログを開始します (クラスタの場合は、プライマリノードから開始するだけで十分です)。

2. アップグレード後に、MRAログインを試行するか、破損した機能をテストします

3. 障害が発生するまで待ってから、Expressway-EおよびCサーバの診断ログを停止します ( クラスタの場合は、クラスタの各ノードから個別にログを収集してください )

4. [コラボレーションソリューションアナライザツール](#)でログをアップロードして分析する

5. 問題が発生すると、影響を受ける各サーバについて、この変更に関連する最新の警告とエラーの行がピックアップされます

The screenshot shows the Cisco Collaboration Solutions Analyzer interface. The main content area displays a diagnostic overview for the file 'diagnostic\_log\_vcsc\_2022-07-11\_17\_33\_18-DifferentCA-8443.tar.gz'. The interface includes a search bar, a sidebar with navigation options (Home, Tools, Log Analyzer, Upload files, Diagnostics, Analysis), and a main panel with tabs for 'Issues found', 'No issue', 'Not applicable', 'Missing information', and 'Potential problem'. The 'Issues found' tab is active, showing a list of issues. The first issue is a warning: 'Duplicate search rule for same protocol which may trigger 2 invites on the targets'. The second issue is a detected alarm: 'Detected alarms in Expressway'. The third issue is a warning: 'Server failed to verify certificate causing TLS issues'. The fourth issue is a warning: 'Certificates expired causing TLS failures and service issues'. The fifth issue is a defect: 'Traffic Server Enforces Certificate Validation of UCM/IMP/Unity nodes for MRA services [CSCw69661]'. This defect is expanded to show related documentation, description, condition, further information, action, and a log snippet. The description states: 'The tomcat[-ECDSA] certificate of the following CUCM / IMP / Unity nodes is not trusted by the Expressway-C: cucm.steven.lab, 10.48.36.215. This leads to MRA login issues.' The condition states: 'Expressway-C X14.2 and higher versions running MRA services are affected.' The further information states: 'Starting with version X14.2 and higher (due to CSCw69661), the Expressway-C traffic server will do a TLS certificate check on the CUCM / IMP / Unity tomcat[-ECDSA] certificates irrespective of the configuration of TLS Verify Mode set when discovering each of those servers.' The action states: '1. Update the Expressway-C trust store with the CA certificates that signed the tomcat[-ECDSA] certificates of CUCM / IMP / Unity nodes. 2. Make sure that the SAN entries of the tomcat certificates contain the IP or FQDN (as shown from the log snippet below) of the respective servers how they are announced over. If you are not able to update the certificates or trust store immediately, you can also apply the workaround on the CLI of the Expressway-C with the following command: xConfiguration EdgeConfigServer VerifyOriginServer: Off'. The log snippet shows the following lines: '2022-07-11T19:33:08.748+02:00 vcsc:traffic\_server[3936]: [ET\_MET #] WARNING: Core server certificate verification failed for (10.48.36.215). Action=terminate Error=self signed certificate in certificate chain server=10.48.36.215(10.48.36.215) depth=1 2022-07-11T19:33:08.748+02:00 vcsc:traffic\_server[3936]: [ET\_MET #] ERROR: SSL connection failed for "10.48.36.215": error:1416F080:SSL routines:tls\_process\_server\_certificate:certificate verify failed 2022-07-11T19:33:08.168+02:00 vcsc:traffic\_server[3936]: [ET\_MET 1] WARNING: Core server certificate verification failed for (cucm.steven.lab). Action=terminate Error=self signed certificate in certificate chain server=cucm.steven.lab(10.48.36.215) depth=1 2022-07-11T19:33:08.168+02:00 vcsc:traffic\_server[3936]: [ET\_MET 1] ERROR: SSL connection failed for "cucm.steven.lab": error:1416F080:SSL routines:tls\_process\_server\_certificate:certificate verify failed'

## CA診断シグニチャ

The screenshot shows the Cisco Collaboration Solutions Analyzer interface. The main content area displays a diagnostic overview for the file 'diagnostic\_log\_vcsc\_2022-07-11\_17\_49\_11-ConnectCAwithIPonCUCM.tar.gz'. The interface includes a search bar, a sidebar with navigation options (Home, Tools, Log Analyzer, Upload files, Diagnostics, Analysis), and a main panel with tabs for 'Issues found', 'No issue', 'Not applicable', 'Missing information', and 'Potential problem'. The 'Issues found' tab is active, showing a list of issues. The first issue is a warning: 'Duplicate search rule for same protocol which may trigger 2 invites on the targets'. The second issue is a detected alarm: 'Detected alarms in Expressway'. The third issue is a warning: 'Server failed to verify certificate causing TLS issues'. The fourth issue is a warning: 'Certificates expired causing TLS failures and service issues'. The fifth issue is a defect: 'Traffic Server Enforces Certificate Validation of UCM/IMP/Unity nodes for MRA services [CSCw69661]'. This defect is expanded to show related documentation, description, condition, further information, action, and a log snippet. The description states: 'The tomcat[-ECDSA] certificate of the following CUCM / IMP / Unity nodes is not trusted by the Expressway-C: 10.48.36.215. This leads to MRA login issues.' The condition states: 'Expressway-C X14.2 and higher versions running MRA services are affected.' The further information states: 'Starting with version X14.2 and higher (due to CSCw69661), the Expressway-C traffic server will do a TLS certificate check on the CUCM / IMP / Unity tomcat[-ECDSA] certificates irrespective of the configuration of TLS Verify Mode set when discovering each of those servers.' The action states: '1. Update the Expressway-C trust store with the CA certificates that signed the tomcat[-ECDSA] certificates of CUCM / IMP / Unity nodes. 2. Make sure that the SAN entries of the tomcat certificates contain the IP or FQDN (as shown from the log snippet below) of the respective servers how they are announced over. If you are not able to update the certificates or trust store immediately, you can also apply the workaround on the CLI of the Expressway-C with the following command: xConfiguration EdgeConfigServer VerifyOriginServer: Off'. The log snippet shows the following lines: '2022-07-11T19:49:01.513+02:00 vcsc:traffic\_server[3936]: [ET\_MET 2] WARNING: SKI (10.48.36.215) not in certificate. Action=terminate server=10.48.36.215(10.48.36.215) 2022-07-11T19:49:01.513+02:00 vcsc:traffic\_server[3936]: [ET\_MET 2] ERROR: SSL connection failed for "10.48.36.215": error:1416F080:SSL routines:tls\_process\_server\_certificate:certificate verify failed'

## SNI診断シグニチャ

## 解決方法

長期的な解決策は、TLS検証が正常に機能していることを確認することです。実行するアクションは、表示される警告メッセージによって異なります。

WARNING: Core server certificate verification failed for (<server-FQDN-or-IP>)が表示された場合 Action=Terminate Error=self signed certificate server=cucm.steven.lab(10.48.36.215) depth=x」というメッセージが表示された場合は、それに応じてExpressway-Cサーバの信頼ストアを更新する必要があります。この証明書に署名したCAチェーン(depth > 0)を使用するか、Maintenance > Security > Trusted CA Certificateで自己署名証明書(depth = 0)を使用します。この操作は、クラスタ内のすべてのサーバで必ず実行してください。別のオプションとして、Expressway-C信頼ストア上の既知のCAによってリモート証明書に署名する方法があります。

---

 注：Expresswayでは、Cisco Bug ID [CSCwa12905](#)のように、同じ共通名(CN)を持つ2つの異なる（たとえば自己署名された）証明書をExpresswayの信頼ストアにアップロードすることはできません。これを修正するには、CA署名付き証明書に移動するか、CUCMをバージョン14にアップグレードします。これで、TomcatとCallManagerで同じ（自己署名）証明書を再利用できます。

---

WARNING: SNI (<server-FQDN-or-IP>) not in certificateメッセージが表示された場合は、提示された証明書にこのサーバのFQDNまたはIPが含まれていないことを示しています。この情報を含むように証明書を適合させることも、設定を変更して（CUCMのシステム>サーバのように）、Expressway-Cサーバの設定を更新してアカウントに反映させることもできます。

## 関連情報

短期的な解決策は、文書化されている回避策を適用して、X14.2.0より前の動作にフォールバックすることです。新しく導入されたコマンドを使用して、Expressway-Cサーバノード上のCLIを介してこれを実行できます。

```
xConfiguration EdgeConfigServer VerifyOriginServer: Off
```

これは、

## 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。