

# Fehlerbehebung: Expressway Traffic Server Certificate Verification for MRA Services Introduced by CSCwc69661/CSCwa25108

## Inhalt

[Einleitung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Hintergrundinformationen](#)

[Vertrauenswürdige Zertifizierungsstellenkette](#)

[SAN- oder CN-Prüfung](#)

[Verhaltensänderung](#)

[Versionen unter X14.2.0](#)

[Versionen von X14.2.0 und höher](#)

[Fehlerbehebung](#)

[1. CA, die das Remote-Zertifikat signiert hat, ist nicht vertrauenswürdig](#)

[2. Die Verbindungsadresse \(FQDN oder IP\) ist im Zertifikat nicht enthalten.](#)

[Einfache Validierung](#)

[Lösung](#)

## Einleitung

Dieses Dokument beschreibt die Verhaltensänderung in den Expressway-Versionen X14.2.0 und höher, die mit der Cisco Bug-ID [CSCwc69661](#) oder der Cisco Bug-ID [CSCwa25108](#) verknüpft sind.

## Voraussetzungen

### Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- Expressway-Basiskonfiguration
- Grundlegende MRA-Konfiguration

### Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf Cisco Expressway mit Version X14.2 und höher.

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten

Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle verstehen.

## Hintergrundinformationen

Mit dieser Verhaltensänderung, die durch die Cisco Bug-ID [CSCwvc69661](#) gekennzeichnet ist oder Cisco Bug-ID [CSCwa25108](#) führt der Datenverkehrsserver auf der Expressway-Plattform eine Zertifikatsüberprüfung des Cisco Unified Communication Manager (CUCM), der Cisco Unified Instant Messaging & Presence (IM&P)- und Unity-Serverknoten für die Mobile and Remote Access (MRA)-Dienste durch. Diese Änderung kann nach einem Upgrade Ihrer Expressway-Plattform zu MRA-Anmeldefehlern führen.

Hypertext Transfer Protocol Secure (HTTPS) ist ein sicheres Kommunikationsprotokoll, das Transport Layer Security (TLS) verwendet, um die Kommunikation zu verschlüsseln. Er erstellt diesen sicheren Kanal mithilfe eines TLS-Zertifikats, das im TLS-Handshake ausgetauscht wird. Auf diese Weise werden zwei Ziele erreicht: Authentifizierung (um zu wissen, mit wem der Remote-Teilnehmer verbunden ist) und Datenschutz (Verschlüsselung). Die Authentifizierung schützt vor Man-in-the-Middle-Angriffen, und der Datenschutz verhindert, dass Angreifer die Kommunikation belauschen und manipulieren.

Die TLS-Verifizierung (Zertifikat) wird im Hinblick auf die Authentifizierung durchgeführt und ermöglicht Ihnen sicherzustellen, dass Sie mit dem richtigen Remote-Teilnehmer verbunden sind. Die Prüfung besteht aus zwei Einzelposten:

1. CA-Kette (Trusted Certificate Authority)
2. Betreff Alternativer Name (SAN) oder Gemeinsamer Name (CN)

## Vertrauenswürdige Zertifizierungsstellenkette

Damit Expressway-C dem von CUCM/IM&P/Unity gesendeten Zertifikat vertraut, muss es in der Lage sein, eine Verbindung von diesem Zertifikat zu einer übergeordneten Zertifizierungsstelle (Certification Authority, CA) herzustellen, der es vertraut. Eine solche Verbindung, eine Hierarchie von Zertifikaten, die ein Entitätszertifikat mit einem Stammzertifikat der Zertifizierungsstelle verknüpft, wird als Vertrauenskette bezeichnet. Um eine solche Vertrauenskette überprüfen zu können, enthält jedes Zertifikat zwei Felder: Aussteller (oder 'Ausgestellt von') und Betreff (oder 'Ausgestellt an').

Serverzertifikate, z. B. das Zertifikat, das CUCM an Expressway-C sendet, weisen im Feld "Subject" (Betreff) in der Regel ihren FQDN (Fully Qualified Domain Name) in der CN auf:

```
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
```

Beispiel für ein Serverzertifikat für CUCM "cucm.vngtp.lab". Es verfügt über den FQDN im CN-Attribut des Felds "Subject" (Betreff) zusammen mit anderen Attributen wie Land (C), Bundesland (ST), Standort (L), ... Es ist auch zu sehen, dass das Serverzertifikat von einer Zertifizierungsstelle mit der Bezeichnung vngtp-ACTIVE-DIR-CA ausgegeben wird.

Hochrangige Zertifizierungsstellen (Root-Zertifizierungsstellen) können ebenfalls ein Zertifikat ausstellen, um sich selbst zu identifizieren. In einem solchen Stammzertifikat der

Zertifizierungsstelle sehen wir, dass Aussteller und Betreff denselben Wert haben:

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

Es handelt sich um ein von einer Stammzertifizierungsstelle ausgestelltes Zertifikat, das sich selbst identifiziert.

In einer typischen Situation stellen die Stammzertifizierungsstellen keine Serverzertifikate aus. Stattdessen stellen sie Zertifikate für andere Zertifizierungsstellen aus. Diese anderen CAs werden dann als zwischengeschaltete CAs bezeichnet. Zwischengeschaltete Zertifizierungsstellen können ihrerseits Serverzertifikate oder Zertifikate für andere zwischengeschaltete Zertifizierungsstellen direkt ausstellen. Es kann vorkommen, dass ein Serverzertifikat von der zwischengeschalteten CA 1 ausgestellt wird, die wiederum ein Zertifikat von der zwischengeschalteten CA 2 erhält usw. Bis schließlich die zwischengeschaltete Zertifizierungsstelle ihr Zertifikat direkt von der Stamm-Zertifizierungsstelle erhält:

```
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
```

Damit Expressway-C dem von CUCM gesendeten Serverzertifikat vertrauen kann, muss es nun in der Lage sein, die Vertrauenskette von diesem Serverzertifikat bis hin zu einem Stammzertifikat der Zertifizierungsstelle zu erstellen. Dazu müssen wir das Root-CA-Zertifikat und alle dazwischen liegenden CA-Zertifikate (falls vorhanden, was nicht der Fall ist, wenn die Root-CA direkt das Server-Zertifikat von CUCM ausgestellt hätte) in den Trust Store von Expressway-C hochladen.

**Anmerkung:** Obwohl die Felder "Issuer" und "Subject" (Betreff) leicht für Menschen lesbar zu erstellen sind, verwendet der CUCM diese Felder im Zertifikat nicht. Stattdessen werden die Felder 'X509v3 Authority Key Identifier' und 'X509v3 Subject Key Identifier' zum Erstellen der Vertrauenskette verwendet. Diese Schlüssel enthalten Bezeichner für Zertifikate, die genauer sind als die Felder Betreff/Aussteller: Es können 2 Zertifikate mit den gleichen Betreff-/Ausstellerfeldern vorhanden sein, aber eines davon ist abgelaufen und eines ist noch gültig. Beide verfügen über einen anderen X509v3-Subject Key Identifier, sodass der CUCM weiterhin die korrekte Vertrauenskette ermitteln kann.

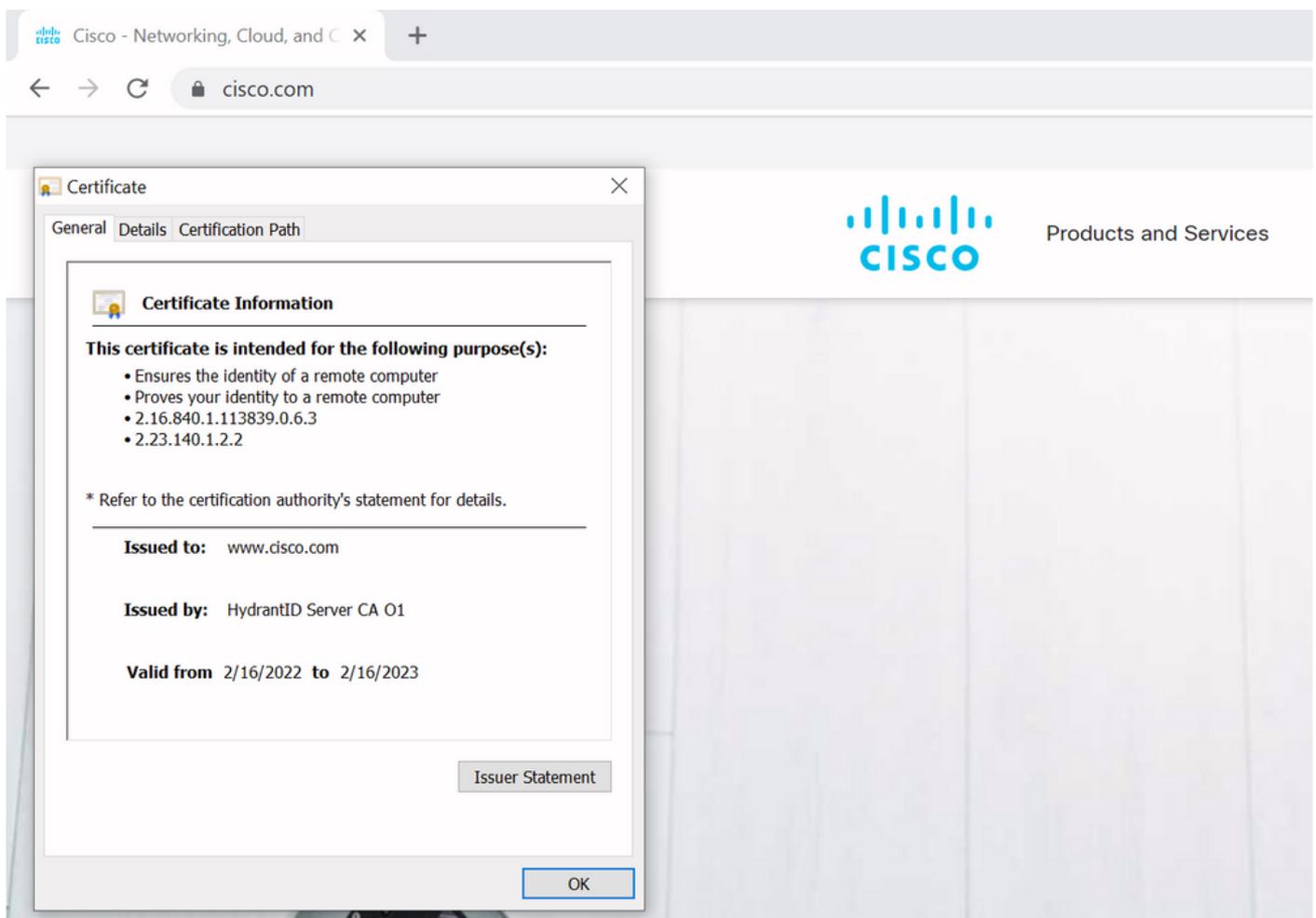
Dies ist bei Expressway nicht der Fall, allerdings gemäß Cisco Bug-ID [CSCwa12905](#), und es ist nicht möglich, zwei verschiedene (z. B. selbstsignierte) Zertifikate in den Trust Store von Expressway hochzuladen, die denselben Common Name (CN) haben. Der Weg, dies zu korrigieren, besteht darin, Zertifikate mit CA-Signatur zu verwenden oder verschiedene Common Names zu verwenden, oder zu sehen, dass immer dasselbe Zertifikat verwendet wird (möglicherweise durch die Funktion zur Wiederverwendung von Zertifikaten in CUCM

14).

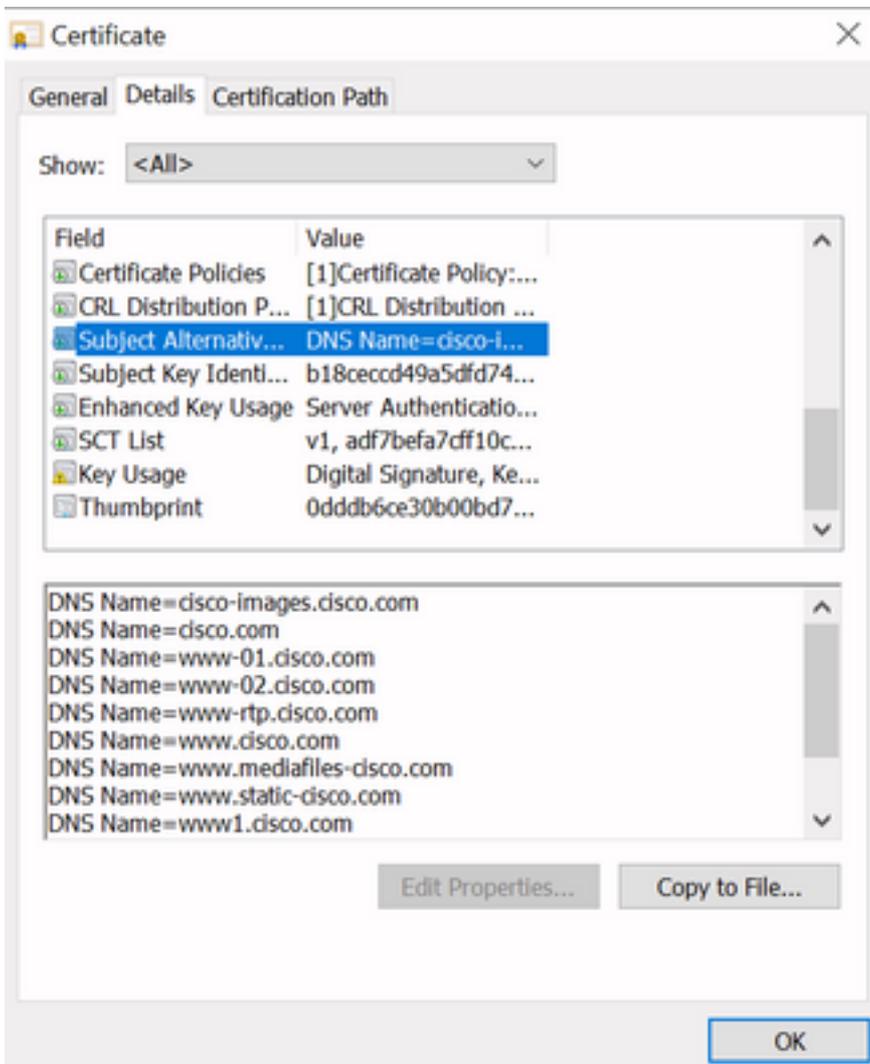
## SAN- oder CN-Prüfung

In Schritt 1 wird der Vertrauensspeicher ausgecheckt. Jeder Benutzer, der über ein Zertifikat verfügt, das von einer Zertifizierungsstelle im Vertrauensspeicher signiert wurde, ist dann jedoch gültig. Dies reicht eindeutig nicht aus. Daher gibt es eine zusätzliche Überprüfung, die bestätigt, dass der Server, mit dem Sie eine Verbindung herstellen, tatsächlich der richtige ist. Dies erfolgt auf der Grundlage der Adresse, für die der Antrag gestellt wurde.

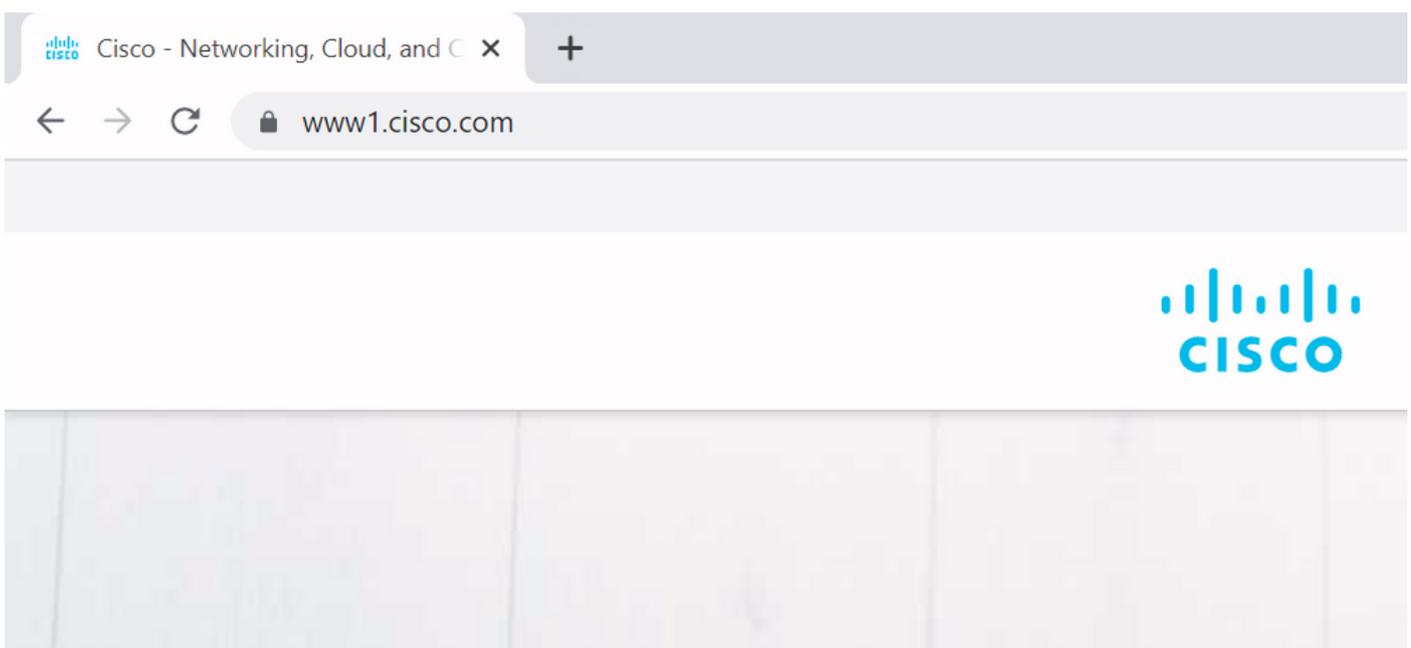
Die gleiche Art von Bedienung geschieht in Ihrem Browser, so lassen Sie uns dies durch ein Beispiel zu betrachten. Wenn Sie zu <https://www.cisco.com> navigieren, sehen Sie ein Sperrsymbol neben der eingegebenen URL. Dies bedeutet, dass es sich um eine vertrauenswürdige Verbindung handelt. Dies basiert sowohl auf der CA Trust Chain (aus dem ersten Abschnitt) als auch auf der SAN- oder CN-Prüfung. Wenn wir das Zertifikat öffnen (über den Browser durch einen Klick auf das Sperrsymbol), sehen Sie, dass der Common Name (siehe Feld "Ausgestellt an:") auf [www.cisco.com](https://www.cisco.com) gesetzt ist und genau der Adresse entspricht, mit der wir uns verbinden wollten. Auf diese Weise kann sichergestellt werden, dass eine Verbindung zum richtigen Server hergestellt wird (da wir der Zertifizierungsstelle vertrauen, die das Zertifikat signiert hat und die eine Überprüfung durchführt, bevor sie das Zertifikat verteilt).



Wenn wir uns die Details des Zertifikats ansehen, und insbesondere die SAN-Einträge, stellen wir fest, dass dies ebenso wie einige andere FQDNs wiederholt werden:



Dies bedeutet, dass, wenn wir z. B. eine Verbindung mit <https://www1.cisco.com> anfordern, diese auch als sichere Verbindung angezeigt wird, da sie in den SAN-Einträgen enthalten ist.



Wenn wir jedoch nicht zu <https://www.cisco.com> navigieren, sondern direkt zu der IP-Adresse (<https://72.163.4.161>), dann wird keine sichere Verbindung angezeigt, da sie der Zertifizierungsstelle vertraut, die sie signiert hat, aber das uns vorgelegte Zertifikat enthält nicht die Adresse (72.163.4.161), mit der wir uns zum Server verbunden haben.

The image shows a browser window with a 'Privacy error' tab and a 'Not secure' warning. The address bar shows 'https://72.163.4.161'. To the left, a Command Prompt window displays the output of an nslookup command for 'cisco.com', showing the server as 'dns-aer1.cisco.com' and the IP address as '72.163.4.161'. To the right, the browser displays a warning: 'Your connection is not private'. It states: 'Attackers might be trying to steal your information from 72.163.4.161 (for example, passwords, messages, or credit cards). Learn more'. Below this, it says 'NET:ERR\_CERT\_COMMON\_NAME\_INVALID'. A button 'To get Chrome's highest level of security, turn on enhanced protection' is visible. At the bottom, it says '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.' and 'Proceed to 72.163.4.161 (unsafe)'.

Im Browser können Sie dies umgehen. Es handelt sich jedoch um eine Einstellung, die Sie für TLS-Verbindungen aktivieren können, sodass eine Umgehung nicht zulässig ist. Daher ist es wichtig, dass Ihre Zertifikate die richtigen CN- oder SAN-Namen enthalten, die der Remote-Teilnehmer verwenden möchte, um eine Verbindung herzustellen.

## Verhaltensänderung

MRA-Services sind stark von mehreren HTTPS-Verbindungen über die Expressways zu den CUCM-/IM&P-/Unity-Servern abhängig, um sich ordnungsgemäß zu authentifizieren und die richtigen Informationen für den angemeldeten Client zu sammeln. Diese Kommunikation findet in der Regel über die Ports 8443 und 6972 statt.

## Versionen unter X14.2.0

In Versionen unter X14.2.0 hat der Datenverkehrsserver auf Expressway-C, der diese sicheren HTTPS-Verbindungen verarbeitet, das vom Remote-Ende vorgelegte Zertifikat nicht überprüft. Dies könnte zu Man-in-the-Middle-Angriffen führen. In der MRA-Konfiguration gibt es eine Option für die TLS-Zertifikatsüberprüfung durch die Konfiguration von "TLS Verify Mode" zu "On", wenn Sie entweder CUCM-/IM&P-/Unity-Server unter "**Configuration**" > "**Unified Communications**" > "**Unified CM-Server**" / "**IM and Presence Service Nodes**" / "**Unity Connection-Server**" hinzufügen würden. Die Konfigurationsoption und das entsprechende Informationsfeld sind als Beispiel dargestellt. Dieses Feld gibt an, dass es den FQDN oder die IP im SAN sowie die Gültigkeit des Zertifikats überprüft und ob es von einer vertrauenswürdigen Zertifizierungsstelle signiert wird.



## Unified CM servers

You are here: [Configuration](#)

Unified CM server lookup	
Unified CM publisher address	cucmpub.vngtp.lab
Username	* administrator
Password	* .....
TLS verify mode	On
Deployment	Default deployment
AES GCM support	Off
SIP UPDATE for session refresh	Off
ICE Passthrough support	Off

Save Delete Cancel

**Information**

If TLS verify mode is enabled, the Unified CM system's FQDN or IP address must be contained within the X.509 certificate presented by that system (in either the Subject Common Name or the Subject Alternative Name attributes of the certificate). The certificate itself must also be valid and signed by a trusted certificate authority.

**Default: On**

Diese Überprüfung des TLS-Zertifikats erfolgt jedoch nur bei der Erkennung der CUCM-/IM&P-/Unity-Server und nicht zum Zeitpunkt der MRA-Anmeldung. Ein erster Nachteil dieser Konfiguration besteht darin, dass sie nur für die Herausgeberadresse verifiziert wird, die Sie hinzufügen. Es wird nicht geprüft, ob das Zertifikat auf den Teilnehmerknoten korrekt eingerichtet wurde, da es die Teilnehmerknoteninformationen (FQDN oder IP) aus der Datenbank des Herausgeberknotens abrufen. Ein zweiter Nachteil dieser Konfiguration besteht darin, dass sich die den MRA-Clients als Verbindungsinformationen gemeldeten Daten von der Herausgeberadresse unterscheiden können, die in der Expressway-C-Konfiguration angegeben wurde. Beispielsweise könnten Sie auf CUCM unter **System > Server** dem Server eine IP-Adresse (z. B. 10.48.36.215) ankündigen, die dann von den MRA-Clients (über die proxidierte Expressway-Verbindung) verwendet wird. Sie könnten CUCM auf Expressway-C jedoch mit dem FQDN von cucm.steven.lab hinzufügen. Nehmen wir also an, dass das Tomcat-Zertifikat von CUCM cucm.steven.lab als SAN-Eintrag, aber nicht die IP-Adresse enthält, dann ist die Erkennung mit

'TLS Verify Mode' auf 'On' erfolgreich, aber die tatsächliche Kommunikation von den MRA-Clients kann auf einen anderen FQDN oder eine andere IP abzielen und somit die TLS-Überprüfung nicht bestehen.

## Versionen von X14.2.0 und höher

Ab der Version X14.2.0 führt der Expressway-Server die TLS-Zertifikatsüberprüfung für jede einzelne HTTPS-Anforderung durch, die über den Datenverkehrsserver erfolgt. Das bedeutet, dass dies auch dann geschieht, wenn der TLS-Verifizierungsmodus während der Erkennung der CUCM-/IM&P-/Unity-Knoten auf "Aus" gesetzt wird. Wenn die Verifizierung nicht erfolgreich ist, wird der TLS-Handshake nicht abgeschlossen, und die Anforderung schlägt fehl. Dies kann zum Verlust von Funktionen führen, z. B. Redundanz- oder Failover-Probleme oder vollständige Anmeldefehler. Auch wenn 'TLS Verify Mode' auf 'On' gesetzt ist, garantiert es nicht, dass alle Verbindungen gut funktionieren, wie im Beispiel weiter unten beschrieben.

Die genauen Zertifikate, die Expressway gegenüber den CUCM-/IM&P-/Unity-Knoten überprüft, sind im Abschnitt des [MRA-Leitfadens](#) aufgeführt.

Neben der Standardeinstellung für die TLS-Verifizierung wurde in X14.2 auch eine Änderung eingeführt, die eine andere Präferenzreihenfolge für die Verschlüsselungsliste ankündigen könnte, die von Ihrem Upgrade-Pfad abhängt. Dies kann nach einem Software-Upgrade zu unerwarteten TLS-Verbindungen führen, da es möglicherweise vor dem Upgrade das Cisco Tomcat- oder Cisco CallManager-Zertifikat vom CUCM (oder einem anderen Produkt, das über ein separates Zertifikat für den ECDSA-Algorithmus verfügt) angefordert hat, nach dem Upgrade jedoch die ECDSA-Variante anfordert (die tatsächlich sicherere Verschlüsselungsvariante als RSA). Die Cisco Tomcat-ECDSA- oder Cisco CallManager-ECDSA-Zertifikate können von einer anderen Zertifizierungsstelle signiert werden oder nur von selbst signierten Zertifikaten (Standard).

Diese Änderung der Reihenfolge der Verschlüsselungspräferenzen ist nicht immer relevant für Sie, da sie vom Upgrade-Pfad abhängt, wie in den [Versionshinweisen](#) zu Expressway X14.2.1 dargestellt. Kurz gesagt können Sie aus **Maintenance > Security > Ciphers** für jede der Chiffrierlisten erkennen, ob sie "ECDHE-RSA-AES256-GCM-SHA384:" voranstellen oder nicht. Ist dies nicht der Fall, wird die neuere ECDSA-Verschlüsselung der RSA-Verschlüsselung vorgezogen. Wenn dies der Fall ist, haben Sie das Verhalten wie zuvor bei RSA, das dann die höhere Präferenz hat.

### 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**).

In diesem Szenario kann die TLS-Überprüfung auf zwei Arten fehlschlagen, die später im Detail behandelt werden:

1. CA, die das entfernte Zertifikat signiert hat, ist nicht vertrauenswürdig

antwort: Selbstsigniertes Zertifikat

b. Zertifikat von unbekannter Zertifizierungsstelle signiert

2. Die Verbindungsadresse (FQDN oder IP) ist im Zertifikat nicht enthalten.

# Fehlerbehebung

Die nächsten Szenarien zeigen ein ähnliches Szenario in einer Laborumgebung, in der die MRA-Anmeldung nach einem Upgrade von Expressway von X14.0.7 auf X14.2 fehlschlug. Sie weisen Gemeinsamkeiten in den Protokollen auf, die Auflösung ist jedoch unterschiedlich. Die Protokolle werden nur durch die Diagnoseprotokollierung erfasst (**Wartung > Diagnose > Diagnoseprotokollierung**), die vor der MRA-Anmeldung gestartet und nach dem Fehler der MRA-Anmeldung beendet wurde. Es wurde keine zusätzliche Debug-Protokollierung aktiviert.

## 1. CA, die das Remote-Zertifikat signiert hat, ist nicht vertrauenswürdig

Das Remote-Zertifikat kann entweder von einer Zertifizierungsstelle signiert werden, die nicht im Vertrauensspeicher des Expressway-C enthalten ist, oder es kann sich um ein selbstsigniertes Zertifikat (im Wesentlichen auch eine Zertifizierungsstelle) handeln, das nicht im Vertrauensspeicher des Expressway-C-Servers hinzugefügt wird.

Im Beispiel hier können Sie beobachten, dass die Anfragen, die an den CUCM gehen (10.48.36.215 - cucm.steven.lab), korrekt auf Port 8443 behandelt werden (200 OK-Antwort), aber es löst einen Fehler (502 Antwort) auf Port 6972 für die TFTP-Verbindung aus.

```
===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="network.http.trafficserver" Level="INFO": Detail="Receive Request" Txn-id="189"
TrackingID="6af9a674-9ebc-41ea-868e-90e7309a758c" Src-ip="127.0.0.1" Src-port="35764" Last-via-
addr="" Msg="GET
http://vcs_control.steven.lab:8443/c3RldmVuLmxhYi9odHRwcy9jdWNTLnN0ZXZlbi5sYWVvODQ0Mw/cucm-
uds/user/emusk/devices HTTP/1.1"
```

```
2022-07-11T18:55:25.917+02:00 vcsc traffic_server[18242]: Event="Request Allowed" Detail="Access
allowed" Reason="In allow list" Username="emusk" Deployment="1" Method="GET"
Request="https://cucm.steven.lab:8443/cucm-uds/user/emusk/devices"
Rule="https://cucm.steven.lab:8443/cucm-uds/user/" Match="prefix" Type="Automatically generated
rule for CUCM server" UTCTime="2022-07-11 16:55:25,916"
```

```
2022-07-11T18:55:25.917+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,916"
Module="network.http.trafficserver" Level="INFO": Detail="Sending Request" Txn-id="189"
TrackingID="6af9a674-9ebc-41ea-868e-90e7309a758c" Dst-ip="10.48.36.215" Dst-port="8443" Msg="GET
/cucm-uds/user/emusk/devices HTTP/1.1"
```

```
2022-07-11T18:55:25.955+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,955"
Module="network.http.trafficserver" Level="INFO": Detail="Receive Response" Txn-id="189"
TrackingID="" Src-ip="10.48.36.215" Src-port="8443" Msg="HTTP/1.1 200 "
```

```
2022-07-11T18:55:25.956+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,955"
Module="network.http.trafficserver" Level="INFO": Detail="Sending Response" Txn-id="189"
TrackingID="" Dst-ip="127.0.0.1" Dst-port="35764" Msg="HTTP/1.1 200 "
```

```
===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="network.http.trafficserver" Level="INFO": Detail="Receive Request" Txn-id="191"
TrackingID="bb0c8492-8c15-4537-a7d1-082dde781dbd" Src-ip="127.0.0.1" Src-port="35766" Last-via-
addr="" Msg="GET
http://vcs_control.steven.lab:8443/c3RldmVuLmxhYi9odHRwcy9jdWNTLnN0ZXZlbi5sYWVvNjk3Mg/CSFemusk.c
nf.xml HTTP/1.1"
```

```
2022-07-11T18:55:26.006+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:26,006"
Module="network.http.trafficserver" Level="INFO": Detail="Sending Request" Txn-id="191"
TrackingID="bb0c8492-8c15-4537-a7d1-082dde781dbd" Dst-ip="10.48.36.215" Dst-port="6972" Msg="GET
```

```

/CSFemusk.cnf.xml HTTP/1.1"
2022-07-11T18:55:26.016+02:00 vscs traffic_server[18242]: UTCTime="2022-07-11 16:55:26,016"
Module="network.http.trafficserver" Level="INFO": Detail="Sending Request" Txn-id="191"
TrackingID="bb0c8492-8c15-4537-a7d1-082dde781dbd" Dst-ip="10.48.36.215" Dst-port="6972" Msg="GET
/CSFemusk.cnf.xml HTTP/1.1"
2022-07-11T18:55:26.016+02:00 vscs 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 vscs 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 vscs traffic_server[18242]: UTCTime="2022-07-11 16:55:26,024"
Module="network.http.trafficserver" Level="INFO": Detail="Sending Response" Txn-id="191"
TrackingID="" Dst-ip="127.0.0.1" Dst-port="35766" Msg="HTTP/1.1 502 connect failed"

```

Der Fehler "Certificate verify failed" (Überprüfung des Zertifikats fehlgeschlagen) zeigt an, dass Expressway-C den TLS-Handshake nicht validieren konnte. Der Grund dafür wird in der Warnzeile angezeigt, da ein selbstsigniertes Zertifikat angezeigt wird. Wenn die Tiefe als 0 angezeigt wird, handelt es sich um ein selbstsigniertes Zertifikat. Wenn die Tiefe größer als 0 ist, bedeutet dies, dass sie über eine Zertifikatskette verfügt und daher von einer unbekanntenen Zertifizierungsstelle signiert wird (aus Sicht von Expressway-C).

Wenn wir die pcap-Datei betrachten, die mit den Zeitstempeln aus den Textprotokollen gesammelt wurde, können Sie sehen, dass CUCM das Zertifikat mit CN als cucm-ms.steven.lab (und cucm.steven.lab als SAN) präsentiert, signiert von steven-DC-CA an den Expressway-C auf Port 8443.

No.	Time	Source	Src port	Destination	Dest port	Protocol	DSCP	VLAN	Length	Info
4693	2022-07-11 16:55:25.916680	10.48.36.46	35622	10.48.36.215	8443	TCP	C50		74	35622 → 8443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=878570435 TSecr=0 WS=128
4693	2022-07-11 16:55:25.916673	10.48.36.46	8443	10.48.36.46	35622	TCP	C50		74	8443 → 35622 [SYN, ACK] Seq=0 Ack=1 Win=28968 Len=0 MSS=1460 SACK_PERM=1 TSval=343633238 TSecr=878570435 WS=128
4694	2022-07-11 16:55:25.917832	10.48.36.46	35622	10.48.36.215	8443	TLSv1.2	C50		66	35622 → 8443 [ACK] Seq=1 Acc=1 Win=64256 Len=0 TSval=878570435 TSecr=343633238
4695	2022-07-11 16:55:25.938856	10.48.36.215	8443	10.48.36.46	35622	TLSv1.2	C50		583	Client Hello
4696	2022-07-11 16:55:25.938208	10.48.36.46	35622	10.48.36.215	8443	TCP	C50		66	35622 → 8443 [ACK] Seq=518 Acc=1449 Win=64256 Len=0 TSval=878570457 TSecr=343633251
4697	2022-07-11 16:55:25.938489	10.48.36.215	8443	10.48.36.46	35622	TLSv1.2	C50		1470	Certificate, Server key Exchange, Server Hello Done
4698	2022-07-11 16:55:25.938419	10.48.36.46	35622	10.48.36.215	8443	TCP	C50		66	35622 → 8443 [ACK] Seq=518 Acc=2853 Win=63488 Len=0 TSval=878570457 TSecr=343633251
4699	2022-07-11 16:55:25.940187	10.48.36.46	35622	10.48.36.215	8443	TLSv1.2	C50		192	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
4700	2022-07-11 16:55:25.943004	10.48.36.215	8443	10.48.36.46	35622	TCP	C50		388	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
4701	2022-07-11 16:55:25.943051	10.48.36.46	35622	10.48.36.215	8443	TCP	C50		66	35622 → 8443 [ACK] Seq=644 Acc=3095 Win=64256 Len=0 TSval=878570461 TSecr=343633256
4702	2022-07-11 16:55:25.943277	10.48.36.46	35622	10.48.36.215	8443	TLSv1.2	C50		2543	Application Data
4703	2022-07-11 16:55:25.943476	10.48.36.215	8443	10.48.36.46	35622	TCP	C50		66	8443 → 35622 [ACK] Seq=3095 Acc=3121 Win=35072 Len=0 TSval=343633256 TSecr=878570462
4707	2022-07-11 16:55:25.954796	10.48.36.215	8443	10.48.36.46	35622	TCP	C50		1514	8443 → 35622 [ACK] Seq=3095 Acc=3121 Win=35072 Len=1448 TSval=343633268 TSecr=878570462 [TCP segment of a reassembled PDU]
4708	2022-07-11 16:55:25.943051	10.48.36.46	35622	10.48.36.215	8443	TCP	C50		66	35622 → 8443 [ACK] Seq=3121 Acc=4543 Win=64256 Len=0 TSval=878570473 TSecr=343633268
4709	2022-07-11 16:55:25.954861	10.48.36.215	8443	10.48.36.46	35622	TLSv1.2	C50		1257	Application Data
4710	2022-07-11 16:55:25.954873	10.48.36.46	35622	10.48.36.215	8443	TCP	C50		66	35622 → 8443 [ACK] Seq=3121 Acc=5734 Win=63488 Len=0 TSval=878570473 TSecr=343633268
4711	2022-07-11 16:55:25.955712	10.48.36.46	35622	10.48.36.215	8443	TLSv1.2	C50		97	Encrypted Alert
4712	2022-07-11 16:55:25.955708	10.48.36.46	35622	10.48.36.215	8443	TCP	C50		66	35622 → 8443 [FIN, ACK] Seq=3152 Acc=5734 Win=64256 Len=0 TSval=878570474 TSecr=343633268
4714	2022-07-11 16:55:25.956123	10.48.36.215	8443	10.48.36.46	35622	TLSv1.2	C50		97	Encrypted Alert
4715	2022-07-11 16:55:25.956170	10.48.36.46	35622	10.48.36.215	8443	TCP	C50		54	35622 → 8443 [RST] Seq=3153 Win=0 Len=0
4716	2022-07-11 16:55:25.956232	10.48.36.215	8443	10.48.36.46	35622	TCP	C50		66	8443 → 35622 [FIN, ACK] Seq=5705 Acc=3153 Win=35072 Len=0 TSval=343633269 TSecr=878570474
4717	2022-07-11 16:55:25.956252	10.48.36.46	35622	10.48.36.215	8443	TCP	C50		54	35622 → 8443 [RST] Seq=3153 Win=0 Len=0

```

Certificates (2425 Bytes)
Certificate Length: 1587
Certificate: 308205df308204c7a00302010201345000012205060503... (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=)
  signedCertificate
    version: v3 (2)
    serialNumber: 0x450000011205060503408084200020000112
    signature (sha1withRSAEncryption)
    issuer: rdnSequence (0)
    validity
    subject: rdnSequence (0)
    subjectPublicKeyInfo
    extensions: 9 items
      Extension (id-ce-extKeyUsage)
      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: cucm.steven.lab
            GeneralName: dNSName (2)
              dNSName: steven.lab
            GeneralName: dNSName (3)
              dNSName: cucm.steven.lab
          Extension (id-ce-subjectKeyIdentifier)
          Extension (id-ce-authorityKeyIdentifier)
          Extension (id-ce-cRLDistributionPoints)
          Extension (id-ce-authorityInfoAccessSyntax)
          Extension (id-ms-certificate-template)
          Extension (id-ms-application-certificate-policies)
    algorithmIdentifier (sha1withRSAEncryption)
    padding: 0
    encrypted: 9fb7f8741637a2a2071efb68f2279cc7cc448478c820...
    Certificate Length: 910
  Certificate: 3082030a30820272a00302010201062176f3f293908044... (id-at-commonName=steven-DC-CA, dc=steven, dc=lab)
Secure Sockets Layer

```

Wenn wir jedoch das auf Port 6972 präsentierte Zertifikat überprüfen, können Sie sehen, dass es sich um ein selbstsigniertes Zertifikat (Issuer ist selbst) mit CN handelt, das als cucm-EC.steven.lab eingerichtet ist. Die Erweiterung -EC gibt an, dass es sich um das ECDSA-Zertifikat handelt, das auf CUCM eingerichtet wurde.

No.	Time	Source	Srv port	Destination	Dst port	Protocol	OSCP	VLAN	Length	Info
4730	2022-07-11 16:55:26.006408	10.40.36.46		11576 10.40.36.215	6972 TCP	C50			74	31576 > 6972 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=878578525 TSecr=0 WS=128
4731	2022-07-11 16:55:26.006853	10.40.36.215		6972 10.40.36.46	31576 TCP	C50			74	6972 > 31576 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=343633320 TSecr=878578525 WS=128
4732	2022-07-11 16:55:26.006892	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			66	31576 > 6972 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=878578525 TSecr=343633320
4733	2022-07-11 16:55:26.007180	10.40.36.46		31576 10.40.36.215	6972 TLSv1.2	C50			583	Client Hello
4734	2022-07-11 16:55:26.016350	10.40.36.215		6972 10.40.36.46	31576 TLSv1.2	C50			1514	Server Hello, Certificate, Server Key Exchange
4735	2022-07-11 16:55:26.016391	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			66	31576 > 6972 [ACK] Seq=518 Ack=1449 Win=64120 Len=0 TSval=878578535 TSecr=343633329
4736	2022-07-11 16:55:26.016408	10.40.36.215		6972 10.40.36.46	31576 TLSv1.2	C50			499	Certificate Request, Server Hello Done
4737	2022-07-11 16:55:26.016419	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			66	31576 > 6972 [ACK] Seq=518 Ack=1882 Win=63744 Len=0 TSval=878578535 TSecr=343633329
4738	2022-07-11 16:55:26.016703	10.40.36.46		31576 10.40.36.215	6972 TLSv1.2	C50			73	Alert (Level: FATAL, Description: Unknown CA)
4739	2022-07-11 16:55:26.016621	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			74	31576 > 6972 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=878578535 TSecr=0 WS=128
4740	2022-07-11 16:55:26.016965	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			66	31576 > 6972 [ACK] Seq=525 Ack=1882 Win=64120 Len=0 TSval=878578535 TSecr=343633329
4741	2022-07-11 16:55:26.016984	10.40.36.215		6972 10.40.36.46	31576 TCP	C50			74	6972 > 31576 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=343633330 TSecr=878578535 WS=128
4742	2022-07-11 16:55:26.017009	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			66	31576 > 6972 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=878578535 TSecr=343633330
4743	2022-07-11 16:55:26.017181	10.40.36.215		6972 10.40.36.46	31576 TCP	C50			66	6972 > 31576 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=343633330 TSecr=878578535
4744	2022-07-11 16:55:26.017121	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			54	31576 > 6972 [RST] Seq=525 Win=0 Len=0
4745	2022-07-11 16:55:26.017218	10.40.36.46		31576 10.40.36.215	6972 TLSv1.2	C50			583	Client Hello
4746	2022-07-11 16:55:26.024226	10.40.36.215		6972 10.40.36.46	31576 TLSv1.2	C50			1514	Server Hello, Certificate, Server Key Exchange
4747	2022-07-11 16:55:26.024265	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			66	31576 > 6972 [ACK] Seq=518 Ack=1449 Win=64120 Len=0 TSval=878578543 TSecr=343633337
4748	2022-07-11 16:55:26.024298	10.40.36.215		6972 10.40.36.46	31576 TLSv1.2	C50			590	Certificate Request, Server Hello Done
4749	2022-07-11 16:55:26.024309	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			66	31576 > 6972 [ACK] Seq=518 Ack=1883 Win=63744 Len=0 TSval=878578543 TSecr=343633337
4750	2022-07-11 16:55:26.024548	10.40.36.46		31576 10.40.36.215	6972 TLSv1.2	C50			73	Alert (Level: Fatal, Description: Unknown CA)
4751	2022-07-11 16:55:26.024647	10.40.36.46		31576 10.40.36.215	6972 TCP	C50			66	31576 > 6972 [RST, ACK] Seq=525 Ack=1883 Win=64120 Len=0 TSval=878578543 TSecr=343633337
4757	2022-07-11 16:55:26.030359	10.40.36.46		31500 10.40.36.215	6972 TCP	C50			74	31500 > 6972 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=878578061 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: 308202820202148083020210207470ee62271e3d1346... (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=BE)
        version: v3 (2)
        serialNumber: 02470ee62271e3d13461d99460a30f5d
        signature (ecdsa-with-SHA384)
        issuer: rdmsquence (8)
        rdnSequence: 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=BE)
        validity
        subject: rdmsquence (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 1: 2.5.29.17 (id-ce-subjectAltName)
              GeneralNames: 1 item
                GeneralName: dnName (2)
                  dnName: cucm.steven.lab
                algorithmIdentifier (ecdsa-with-SHA384)
                padding: 0
                encrypted: 306402020214395d5e8e74570b1171eb49f9a30b6ec0d08...
  TLSv1.2 Record Layer: Handshake Protocol: Server Key Exchange
  
```

Auf CUCM unter Cisco Unified OS-Administration können Sie die vorhandenen Zertifikate unter Sicherheit > Zertifikatsverwaltung einsehen, wie hier gezeigt. Es wird ein anderes Zertifikat für Tomcat und Tomcat-ECDSA angezeigt, bei dem der Tomcat mit CA signiert ist (und von Expressway-C als vertrauenswürdig eingestuft wird), während das Tomcat-ECDSA-Zertifikat selbstsigniert ist und von Expressway-C hier nicht als vertrauenswürdig eingestuft wird.

Certificate	Common Name	Type	Key Type	Distribution	Issued by	Expiration	Description
authZ	AUTHZ_cucm.steven.lab	Self-signed	RSA	cucm.steven.lab	AUTHZ_cucm.steven.lab	07/21/2038	Self-signed certificate generated by system
CallManager	cucm.steven.lab	CA-signed	RSA	cucm.steven.lab	stevenc-oc-ca	07/13/2022	Certificate Signed by stevenc-oc-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	stevenc-oc-ca	Self-signed	RSA	stevenc-oc-ca	stevenc-oc-ca	06/01/2023	Signed Certificate
CallManager-trust	NOMAT-AD-CA	Self-signed	RSA	NOMAT-AD-CA	NOMAT-AD-CA	04/23/2028	Signed Certificate
CallManager-trust	CAP-RTF-002	Self-signed	RSA	CAP-RTF-002	CAP-RTF-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-eb26468	Self-signed	RSA	CAPF-eb26468	CAPF-eb26468	04/12/2020	Signed Certificate
CallManager-trust	ms-AD2-CA-1	Self-signed	RSA	ms-AD2-CA-1	ms-AD2-CA-1	09/11/2024	vmgtp CA
CallManager-trust	CAP-RTF-001	Self-signed	RSA	CAP-RTF-001	CAP-RTF-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	RSA	NOMAT-CA-10	NOMAT-CA-10	08/11/2027	Signed Certificate
CallManager-trust	Cisco_Root_CA_M2	Self-signed	RSA	Cisco_Root_CA_M2	Cisco_Root_CA_M2	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_SUDD_CA	CA-signed	RSA	ACT2_SUDD_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	vmgtp-ACTIVE-DIR-CA	Self-signed	RSA	vmgtp-ACTIVE-DIR-CA	vmgtp-ACTIVE-DIR-CA	02/10/2024	VMGTP-CA
CallManager-trust	Cisco_Root_CA_2048	Self-signed	RSA	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	RSA	Cisco_Manufacturing_CA_SHA2	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	RSA	Cisco_Manufacturing_CA_SHA2	Cisco_Root_CA_M2	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	dcocomics-WONDERWOMAN-CA	Self-signed	RSA	dcocomics-WONDERWOMAN-CA	dcocomics-WONDERWOMAN-CA	09/19/2037	CA-variant
CallManager-trust	CAPF-616421bc	Self-signed	RSA	CAPF-616421bc	CAPF-616421bc	07/12/2025	Self-signed certificate generated by system
CallManager-trust	CAPF-616421bc	Self-signed	RSA	cucm.steven.lab	CAPF-616421bc	07/12/2025	Self-signed certificate generated by system
CallManager-trust	CAP-RTF-002	Self-signed	RSA	CAP-RTF-002	CAP-RTF-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-eb26468	Self-signed	RSA	CAPF-eb26468	CAPF-eb26468	04/12/2020	Signed Certificate
CallManager-trust	CAP-RTF-001	Self-signed	RSA	CAP-RTF-001	CAP-RTF-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	Cisco_Root_CA_M2	Self-signed	RSA	Cisco_Root_CA_M2	Cisco_Root_CA_M2	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_SUDD_CA	CA-signed	RSA	ACT2_SUDD_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	Cisco_Root_CA_2048	Self-signed	RSA	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	RSA	Cisco_Manufacturing_CA_SHA2	Cisco_Root_CA_M2	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	CAPF-616421bc	Self-signed	RSA	CAPF-616421bc	CAPF-616421bc	07/12/2025	Self-signed certificate generated by system
ispac	cucm.steven.lab	Self-signed	RSA	cucm.steven.lab	cucm.steven.lab	07/12/2025	Self-signed certificate generated by system
ispac-trust	cucm.steven.lab	Self-signed	RSA	cucm.steven.lab	cucm.steven.lab	07/12/2025	Trust Certificate
ITLRecovery	ITLRECOVERY_cucm.steven.lab	Self-signed	RSA	cucm.steven.lab	ITLRECOVERY_cucm.steven.lab	02/14/2039	Self-signed certificate generated by system
stevenc	stevenc-oc-ca	CA-signed	RSA	stevenc-oc-ca	stevenc-oc-ca	07/10/2024	Certificate Signed by stevenc-oc-ca
tomcat	cucm-ec.steven.lab	Self-signed	EC	cucm.steven.lab	cucm-ec.steven.lab	07/25/2023	Self-signed certificate generated by system
tomcat-ECDSA	stevenc-oc-ca	Self-signed	RSA	stevenc-oc-ca	stevenc-oc-ca	06/01/2023	Trust Certificate
tomcat-trust	NOMAT-AD-CA	Self-signed	RSA	NOMAT-AD-CA	NOMAT-AD-CA	07/25/2023	Trust Certificate
tomcat-trust	cucm-ec.steven.lab	Self-signed	EC	cucm.steven.lab	cucm-ec.steven.lab	07/25/2023	Trust Certificate
tomcat-trust	stevenc-oc-ca	Self-signed	RSA	stevenc-oc-ca	stevenc-oc-ca	07/10/2024	Trust Certificate
tomcat-trust	cucm-ec.steven.lab	Self-signed	EC	cucm.steven.lab	cucm-ec.steven.lab	07/25/2023	Trust Certificate
tomcat-trust	NOMAT-CA-10	Self-signed	RSA	NOMAT-CA-10	NOMAT-CA-10	08/11/2027	Signed Certificate
tomcat-trust	vmgtp-ACTIVE-DIR-CA	Self-signed	RSA	vmgtp-ACTIVE-DIR-CA	vmgtp-ACTIVE-DIR-CA	02/10/2024	Trust Certificate
tomcat-trust	dcocomics-WONDERWOMAN-CA	Self-signed	RSA	dcocomics-WONDERWOMAN-CA	dcocomics-WONDERWOMAN-CA	09/19/2037	CA Bruno
TVS	cucm.steven.lab	Self-signed	RSA	cucm.steven.lab	cucm.steven.lab	07/12/2025	Self-signed certificate generated by system

2. Die Verbindungsadresse (FQDN oder IP) ist im Zertifikat nicht enthalten.

Neben dem Trust Store überprüft der Datenverkehrsserver auch die Verbindungsadresse, an die der MRA-Client die Anforderung sendet. Wenn Sie z. B. auf dem CUCM unter System > Server

Ihren CUCM mit der IP-Adresse (10.48.36.215) eingerichtet haben, dann meldet der Expressway-C dies dem Client als solches und nachfolgende Anfragen vom Client (über den Expressway-C weitergeleitet) werden auf diese Adresse ausgerichtet.

Wenn diese spezielle Verbindungsadresse nicht im Serverzertifikat enthalten ist, schlägt auch die TLS-Überprüfung fehl, und es wird ein 502-Fehler ausgelöst, der beispielsweise zu einem MRA-Anmeldefehler führt.

```
2022-07-11T19:49:01.472+02:00 vcsc traffic_server[3916]: UTCTime="2022-07-11 17:49:01,472"
Module="network.http.trafficserver" Level="DEBUG": Detail="Receive Request" Txn-id="144"
TrackingID="0a334fa8-41e9-4b97-adf4-e165372c38cb" Src-ip="127.0.0.1" Src-port="30044" Last-via-
addr=""
HTTPMSG:
|GET http://vcs_control.steven.lab:8443/c3RldmVuLmxhYi9odHRwcy8xMC40OC4zNi4yMTUvODQ0Mw/cucm-
uds/user/emusk/devices?max=100 HTTP/1.1
...

2022-07-11T19:49:01.478+02:00 vcsc traffic_server[3916]: UTCTime="2022-07-11 17:49:01,478"
Module="network.http.trafficserver" Level="INFO": Detail="Sending Request" Txn-id="144"
TrackingID="0a334fa8-41e9-4b97-adf4-e165372c38cb" Dst-ip="10.48.36.215" Dst-port="8443" Msg="GET
/cucm-uds/user/emusk/devices?max=100 HTTP/1.1"
2022-07-11T19:49:01.478+02:00 vcsc traffic_server[3916]: UTCTime="2022-07-11 17:49:01,478"
Module="network.http.trafficserver" Level="DEBUG": Detail="Sending Request" Txn-id="144"
TrackingID="0a334fa8-41e9-4b97-adf4-e165372c38cb" Dst-ip="10.48.36.215" Dst-port="8443"
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
Wobei c3RldmVuLmxhYi9odHRwcy8xMC40OC4zNi4yMTUvODQ0Mw (base64 -
https://www.base64decode.org/) zu steven.lab/https/10.48.36.215/8443 übersetzt, was bedeutet,
dass eine Verbindung zu 10.4 hergestellt werden muss 8.36.215 als Verbindungsadresse und
nicht an cucm.steven.lab. Wie in den Paketerfassungen dargestellt, enthält das CUCM-Tomcat-
Zertifikat nicht die IP-Adresse im SAN, sodass der Fehler ausgelöst wird.
```

## Einfache Validierung

Mit den folgenden Schritten können Sie überprüfen, ob sich dieses Verhalten auf einfache Weise ändert:

1. Starten Sie die Diagnoseprotokollierung auf Expressway-E und C-Servern (idealerweise mit aktivierten TCPDumps) über **Wartung > Diagnose > Diagnoseprotokollierung** (bei einem Cluster reicht es aus, ihn vom Primärknoten aus zu starten).
2. Versuchen Sie eine MRA-Anmeldung oder testen Sie die defekte Funktionalität nach dem Upgrade
3. Warten Sie, bis der Fehler behoben ist, und stoppen Sie dann die Diagnoseprotokollierung auf den Expressway-E- und C-Servern (im Falle eines Clusters müssen Sie sicherstellen, dass die Protokolle von jedem einzelnen Knoten des Clusters einzeln erfasst werden).

#### 4. Hochladen und Analysieren der Protokolle mit dem [Collaboration Solution Analyzer-Tool](#)

5. Wenn das Problem auftritt, werden die letzten Warn- und Fehlerzeilen für jeden betroffenen Server erfasst, die sich auf diese Änderung beziehen

The screenshot shows the 'Diagnostic overview' page in the Collaboration Solutions Analyzer Log Analyzer. The interface includes a sidebar with navigation options (Home, Tools, Log Analyzer, Upload files, Diagnostics, Analysis) and a main content area. The main area displays a search bar, a 'Result Category' filter (Call (53), MRA (51), Configuration (39)), and a 'Defects only' toggle. The central pane shows a list of issues under the 'Issues found' tab. The selected issue is 'Traffic Server Enforces Certificate Validation of UCM/IMSP/Unity nodes for MRA services [CSCw69661]'. The detailed view for this issue includes:

- Description:** The tomcat(-ECDSA) certificate of the following CUCM / IMSP / Unity nodes is not trusted by the Expressway-C: cucm.steven.lab, 10.48.36.215. This leads to MRA login issues.
- Condition:** Expressway-C X14.2 and higher versions running MRA services are affected.
- Further information:** Starting with version X14.2 and higher (due to CSCw69661), the Expressway-C traffic server will do a TLS certificate check on the CUCM / IMSP / Unity tomcat(-ECDSA) certificates irrespective of the configuration of TLS Verify Mode set when discovering each of those servers.
- Action:**
  - Update the Expressway-C trust store with the CA certificates that signed the tomcat(-ECDSA) certificates of CUCM / IMSP / Unity nodes.
  - 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.
- Snippet:** Log entries showing warnings and errors related to certificate verification failures.

CA-Diagnosesignatur

This screenshot is similar to the previous one, showing the 'Diagnostic overview' page. The selected issue is 'Server failed to verify certificate causing TLS issues'. The detailed view for this issue includes:

- Description:** The tomcat(-ECDSA) certificate of the following CUCM / IMSP / Unity nodes is not trusted by the Expressway-C: 10.48.36.215. This leads to MRA login issues.
- Condition:** Expressway-C X14.2 and higher versions running MRA services are affected.
- Further information:** Starting with version X14.2 and higher (due to CSCw69661), the Expressway-C traffic server will do a TLS certificate check on the CUCM / IMSP / Unity tomcat(-ECDSA) certificates irrespective of the configuration of TLS Verify Mode set when discovering each of those servers.
- Action:**
  - Update the Expressway-C trust store with the CA certificates that signed the tomcat(-ECDSA) certificates of CUCM / IMSP / Unity nodes.
  - 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.
- Snippet:** Log entries showing warnings and errors related to certificate verification failures.

SNI-Diagnosesignatur

## Lösung

Die langfristige Lösung besteht darin, sicherzustellen, dass die TLS-Verifizierung reibungslos funktioniert. Welche Aktion ausgeführt werden soll, hängt von der angezeigten Warnmeldung ab.

Wenn Sie die *WARNUNG* beobachten: *Fehler bei der Überprüfung des Kernserverzertifikats für (<server-FQDN-or-IP>). Action=Terminate Error=self signed certificate server=cucm.steven.lab(10.48.36.215) depth=x* message, dann müssen Sie den Trust Store auf den Expressway-C-Servern entsprechend aktualisieren. Entweder mit der Zertifizierungsstellenkette, die dieses Zertifikat signiert hat (Tiefe > 0), oder mit dem selbstsignierten Zertifikat (Tiefe = 0) von **Maintenance > Security > Trusted CA Certificate**. Führen Sie diese Aktion auf jedem Server im Cluster aus. Eine weitere Option besteht darin, das Remote-Zertifikat von einer bekannten Zertifizierungsstelle im Expressway-C-Vertrauensspeicher zu signieren.

**Anmerkung: Expressway erlaubt es nicht, zwei verschiedene (z. B. selbstsignierte) Zertifikate in den Trust Store von Expressway hochzuladen, die denselben Common Name (CN) wie die Cisco Bug-ID [CSCwa12905](#) aufweisen. Um dies zu korrigieren, wechseln Sie zu CA-signierten Zertifikaten, oder aktualisieren Sie Ihren CUCM auf Version 14, wo Sie dasselbe (selbstsignierte) Zertifikat wiederverwenden können für Tomcat und CallManager.**

Wenn Sie die *WARNUNG* beobachten: *SNI (<server-FQDN-or-IP>) nicht in der Zertifikatsnachricht angegeben*, dann wird angegeben, dass dieser Server-FQDN oder diese Server-IP nicht im vorgelegten Zertifikat enthalten ist. Sie können entweder das Zertifikat anpassen, um diese Informationen einzubeziehen, oder Sie können die Konfiguration ändern (z. B. auf CUCM auf System > Server, um etwas zu erhalten, das im Serverzertifikat enthalten ist) und dann die Konfiguration auf dem Expressway-C-Server aktualisieren, damit sie berücksichtigt wird.

Die kurzfristige Lösung besteht darin, die beschriebene Problemumgehung anzuwenden, um auf das vorherige Verhalten vor X14.2.0 zurückzugreifen. Sie können dies mithilfe der CLI auf den Expressway-C-Serverknoten mit dem neu eingeführten Befehl ausführen:

```
xConfiguration EdgeConfigServer VerifyOriginServer: Off
```

## Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.