Probleemoplossing Expressway Traffic Server-certificaatverificatie voor MRA-services, geïntroduceerd door CSCwc69661 / CSCwa25108

Inhoud

Inleiding

Dit document beschrijft de gedragsverandering op Expressway-versies van X14.2.0 en hoger die zijn gekoppeld aan Cisco bug-id CSCwc69661 of Cisco bug-id CSCwa25108.

Voorwaarden

Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- Basisconfiguratie snelweg
- MRA basisconfiguratie

Gebruikte componenten

De informatie in dit document is gebaseerd op Cisco Expressway op versie X14.2 en hoger.

De informatie in dit document is gebaseerd op de apparaten in een specifieke

laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u zorgen dat u de potentiële impact van elke opdracht begrijpt.

Achtergrondinformatie

Met deze gedragswijziging die wordt gemarkeerd met Cisco bug ID CSCwc69661 of Cisco-bug-id CSCwa25108 , voert de verkeersserver op het Expressway-platform certificaatverificatie uit van de Cisco Unified Communications Manager (CUCM), Cisco Unified Instant Messaging & Presence (IM&P) en Unity-serverknooppunten voor de Mobile en Remote Access (MRA)-services. Deze verandering kan leiden tot MRA inlogfouten na een upgrade op uw Expressway platform.

HTTPS (Hypertext Transfer Protocol Secure) is een veilig communicatieprotocol dat Transport Layer Security (TLS) gebruikt om de communicatie te versleutelen. Het maakt dit beveiligde kanaal door het gebruik van een TLS-certificaat dat wordt uitgewisseld in de TLS-handdruk. Op die manier worden twee doelen nagestreefd: verificatie (om te weten wie de externe partij is waarmee u verbinding maakt) en privacy (de codering). De authenticatie beschermt tegen man-in-the-middle aanvallen en de privacy voorkomt dat aanvallers kunnen afluisteren en knoeien op de communicatie.

TLS (certificaat) verificatie wordt uitgevoerd in het zicht van authenticatie en u kunt er zeker van zijn dat u hebt verbonden met de juiste externe partij. De controle bestaat uit twee afzonderlijke elementen:

1. Keten van de Trusted Certificate Authority (CA)

2. Onderwerp Alternatieve naam (SAN) of algemene naam (CN)

Trusted CA-keten

Om ervoor te zorgen dat Expressway-C het certificaat vertrouwt dat CUCM / IM&P / Unity verzendt, moet het een link kunnen maken van dat certificaat naar een top level (root) certificeringsinstantie (CA) die het vertrouwt. Zulke een link, een hiërarchie van certificaten die een entiteitscertificaat koppelen aan een wortel CA certificaat, wordt een keten van vertrouwen genoemd. Om een dergelijke vertrouwensketen te kunnen verifiëren, bevat elk certificaat twee velden: Emittent (of 'Afgegeven door') en Onderwerp (of 'Afgegeven aan').

Servercertificaten, zoals het certificaat dat CUCM naar Expressway-C stuurt, hebben in het veld 'Onderwerp' doorgaans hun volledig gekwalificeerde domeinnaam (FQDN) in de GN:

```
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
```

Voorbeeld van een servercertificaat voor CUCM cucm.vngtp.lab. Het heeft het FQDN in de eigenschap CN van het veld Onderwerp samen met andere attributen zoals Land (C), Staat (ST), Plaats (L), ... We kunnen ook zien dat het servercertificaat wordt afgegeven door een CA met de naam vngtp-ACTIVE-DIR-CA.

Top level CA's (root CA's) kunnen ook een certificaat uitgeven om zichzelf te identificeren. In een dergelijk basiscertificaat van CA zien we dat de Emittent en Onderwerp dezelfde waarde hebben:

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

Het is een certificaat dat wordt afgegeven door een wortel CA om zichzelf te identificeren.

In een typische situatie geven root-CA's niet direct servercertificaten uit. In plaats daarvan geven zij certificaten af voor andere CA's. Dergelijke andere CA's worden dan intermediaire CA's genoemd. Intermediaire CA's kunnen op hun beurt rechtstreeks servercertificaten of certificaten voor andere intermediaire CA's afgeven. We kunnen een situatie hebben waarin een servercertificaat wordt afgegeven door tussenliggende CA 1, die op zijn beurt een certificaat krijgt van tussenliggende CA 2 enzovoort. Totdat uiteindelijk tussenliggende CA haar certificaat rechtstreeks van de wortel CA krijgt:

```
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
```

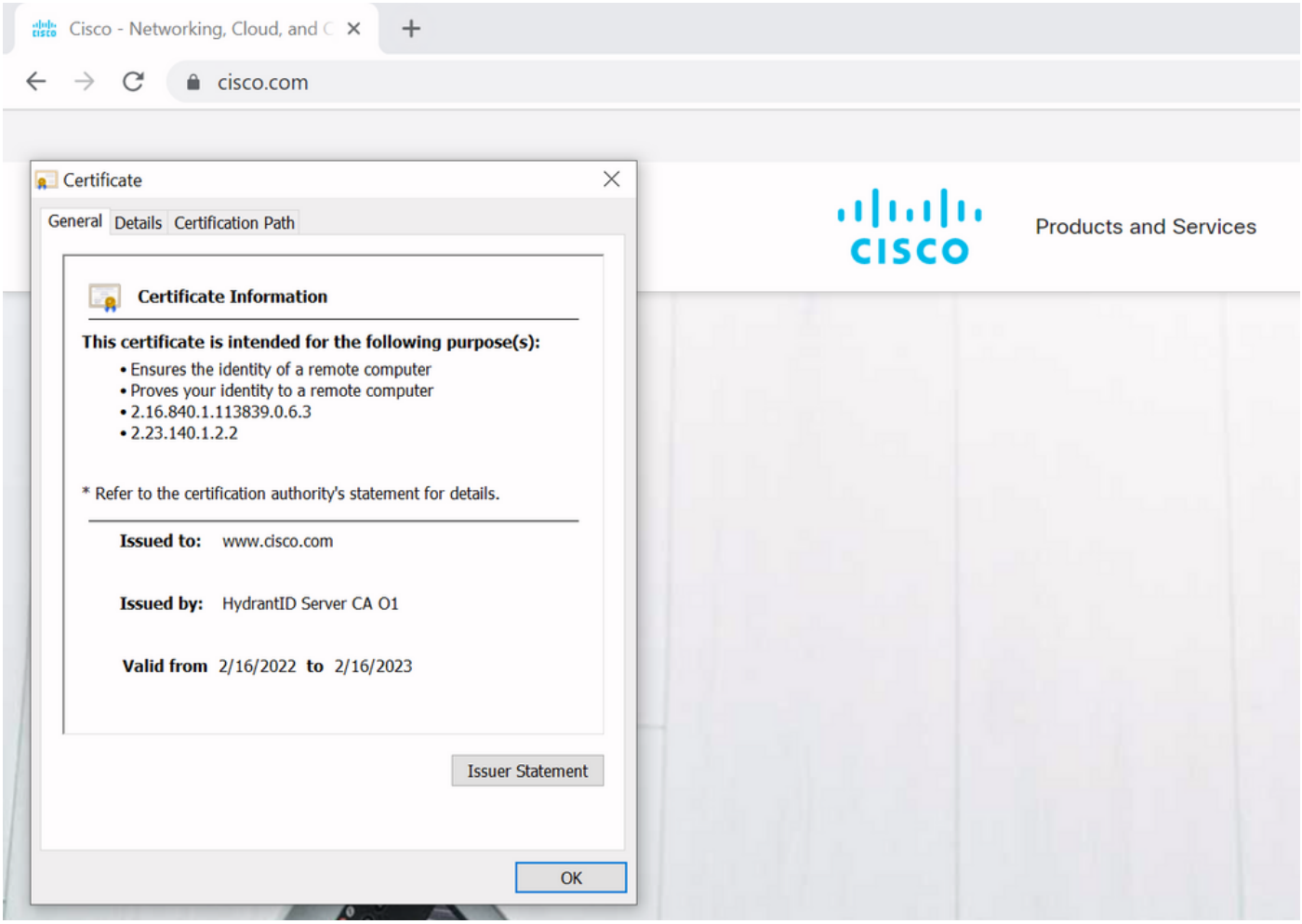
Om Expressway-C te kunnen vertrouwen op het servercertificaat dat CUCM verstuurt, moet het in staat zijn om de keten van vertrouwen van dat servercertificaat tot een wortel CA certificaat op te bouwen. Om dat mogelijk te maken, moeten we het root CA certificaat uploaden en ook alle tussenliggende CA certificaten (als er een zijn, wat niet het geval is als de root CA direct het server certificaat van CUCM zou hebben verstrekt) in de vertrouwensopslag van Expressway-C.

> **Opmerking: Hoewel de velden Emittent en Onderwerp gemakkelijk zijn om de keten van Vertrouwen op een menselijk leesbare manier op te bouwen, gebruikt CUCM deze velden niet in het certificaat. In plaats daarvan gebruikt het de 'X509v3 Authority Key Identifier' en de 'X509v3 subject Key Identifier' velden om de keten van vertrouwen op te bouwen. Deze sleutels bevatten identificatiecodes voor de certificaten die nauwkeuriger zijn dan het gebruik van de velden Onderwerp/Afgevende: er kunnen 2 certificaten zijn met dezelfde velden Onderwerp/Emittent, maar één daarvan is verlopen en één is nog steeds geldig. Ze zouden beide een andere X509v3 Onderwerp Key identifier hebben, zodat CUCM nog steeds de juiste keten van vertrouwen kan bepalen.**
>
> **Dit is echter niet het geval voor Expressway hoewel volgens Cisco bug ID CSCwa12905 en het niet mogelijk is om twee verschillende (zelf-ondertekende bijvoorbeeld) certificaten te uploaden in het vertrouwensarchief van Expressway die dezelfde algemene naam (CN) hebben. De manier om dit te corrigeren, is door CA ondertekende certificaten te gebruiken of verschillende gemeenschappelijke namen te gebruiken voor het of te zien dat het altijd hetzelfde certificaat gebruikt (mogelijk via de functie van het hergebruikscertificaat in CUCM 14).**

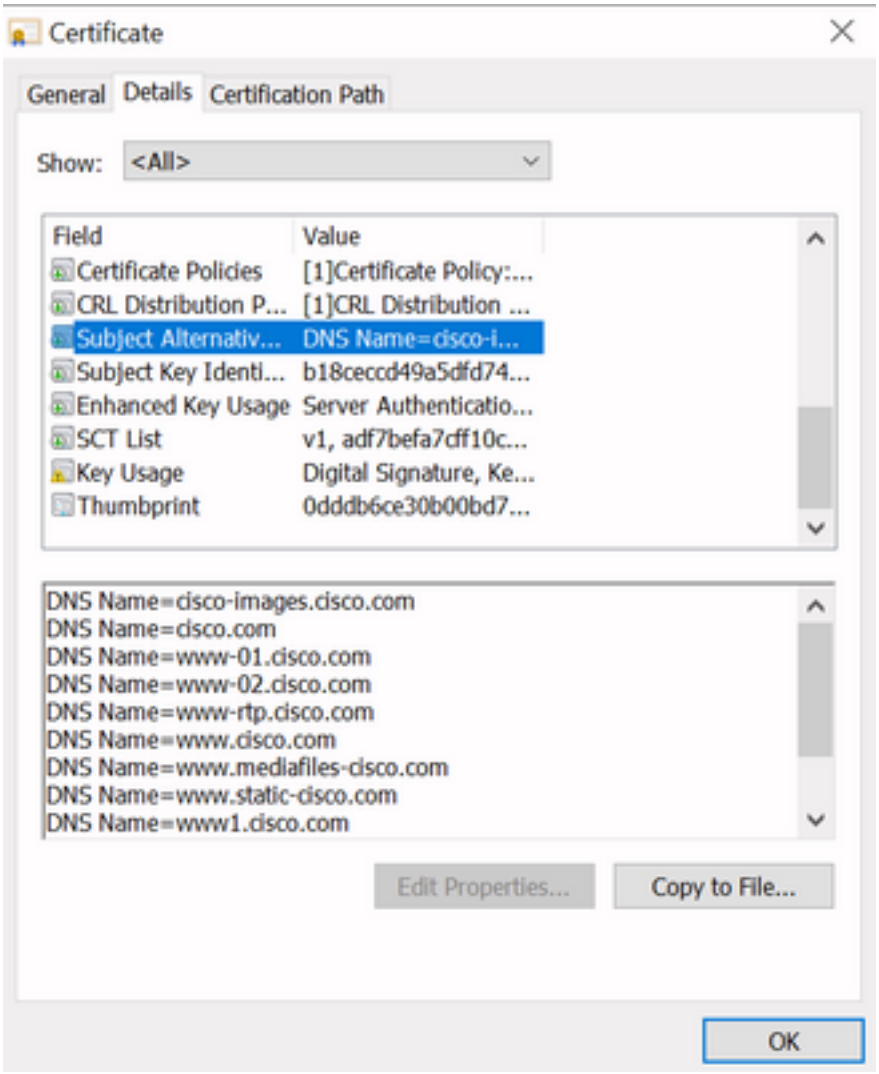
SAN of CN controleren

Stap 1 controleert de trust store, maar iedereen die een certificaat heeft dat is ondertekend door een CA in de trust store zou dan geldig zijn. Dit is duidelijk niet voldoende. Daarom is er een extra controle die bevestigt dat de server waarmee u specifiek verbinding maakt, inderdaad de juiste is. Dit gebeurt op basis van het adres waarvoor het verzoek is ingediend.

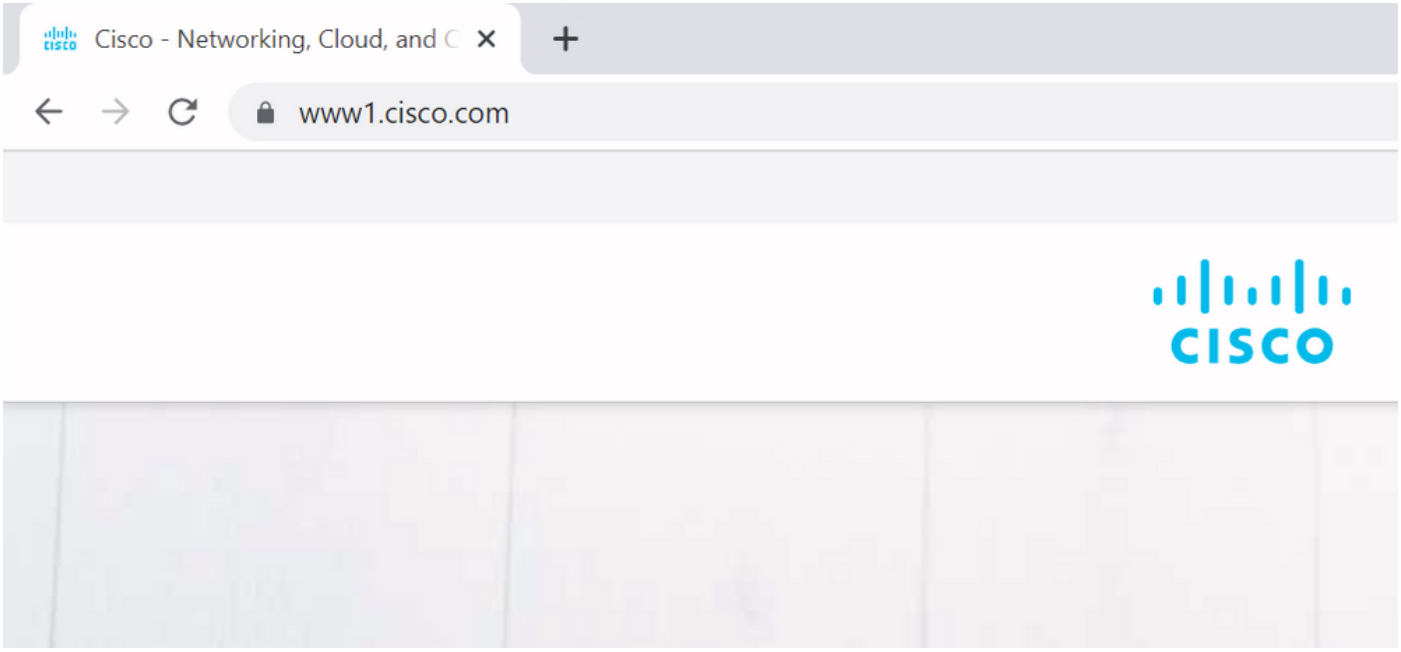
Het zelfde soort operatie gebeurt in uw browser dus laten we dit door een voorbeeld bekijken. Als u naar https://www.cisco.com bladert, ziet u een slotpictogram naast de URL die u hebt ingevoerd en dit betekent dat het een vertrouwde verbinding is. Dit is zowel gebaseerd op de CA-vertrouwensketen (uit eerste sectie) als op de SAN- of CN-controle. Als we het certificaat openen (via de browser door een klik op het slotpictogram), zie je dat de algemene naam (gezien op 'Uitgegeven aan:' veld) is ingesteld op www.cisco.com en dat precies overeenkomt met het adres waarmee we verbinding wilden maken. Op die manier kan het zeker zijn dat we verbinding maken met de juiste server (omdat we vertrouwen op de CA die het certificaat heeft ondertekend en die de verificatie uitvoert voordat het het certificaat uitdeelt).



Wanneer we kijken naar de details van het certificaat en in het bijzonder naar de SAN-vermeldingen, zien we dat hetzelfde wordt herhaald als sommige andere FQDN's:

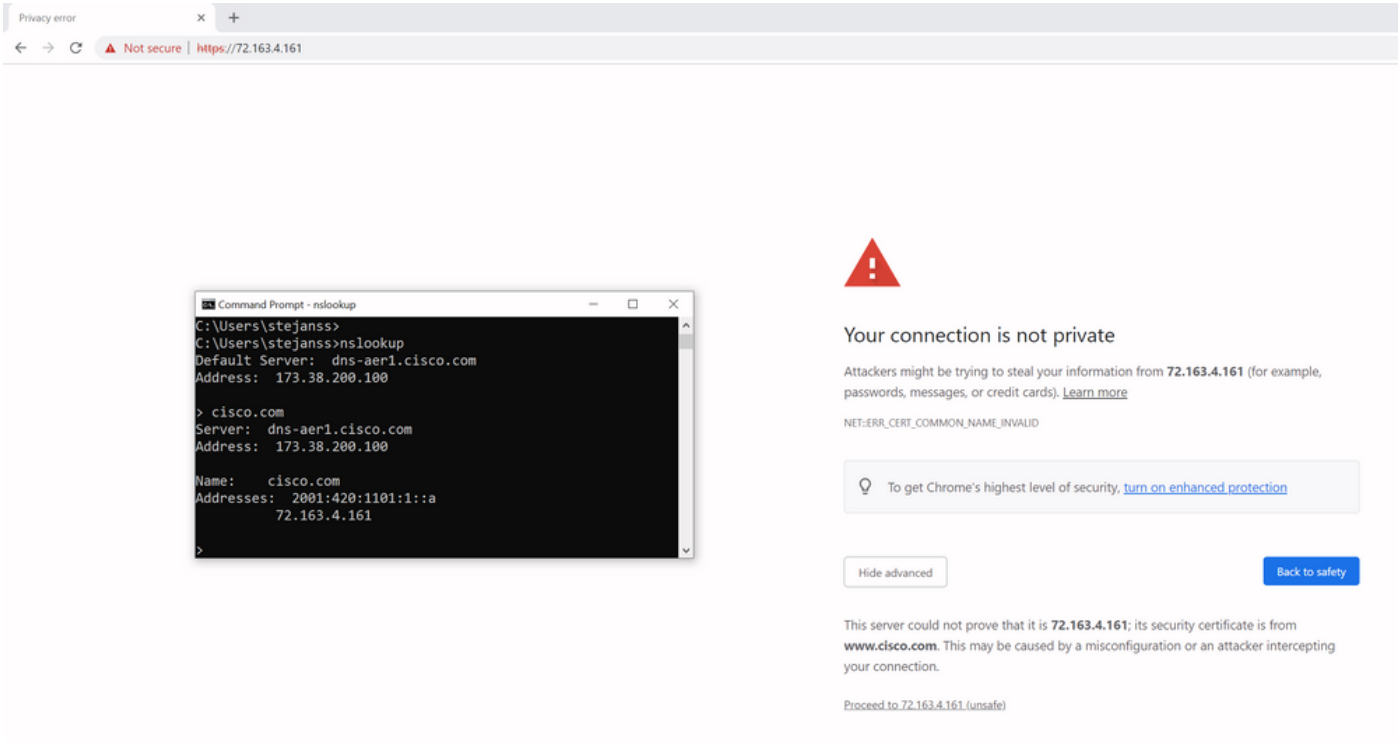


Dit betekent dat wanneer we zouden vragen om verbinding te maken met bijvoorbeeld https://www1.cisco.com, dat het ook als een veilige verbinding zou verschijnen omdat het is opgenomen in de SAN-vermeldingen.



Als we echter niet naar https://www.cisco.com zouden bladeren maar direct naar het IP-adres (https://72.163.4.161), dan wordt er geen beveiligde verbinding weergegeven omdat het wel vertrouwt op de CA die het heeft ondertekend, maar het certificaat dat aan ons wordt aangeboden,

bevat niet het adres (72.163.4.161) dat we gebruikten om verbinding te maken met de server.



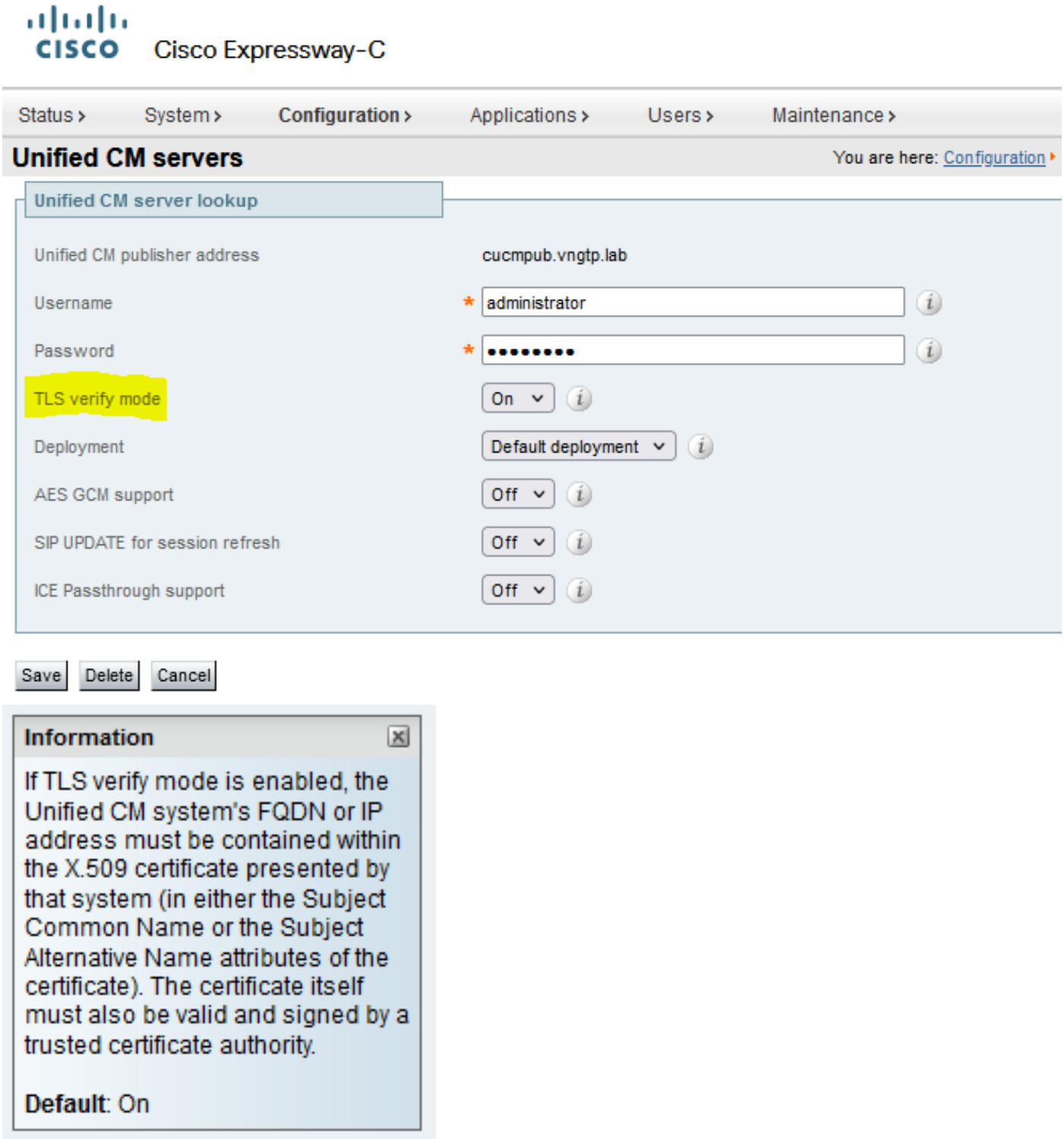
In de browser kunt u dit omzeilen, maar het is een instelling die u kunt inschakelen op TLS-verbindingen dat een omzeiling niet is toegestaan. Daarom is het belangrijk dat uw certificaten de juiste CN- of SAN-namen bevatten die de externe partij wil gebruiken om er verbinding mee te maken.

Gedragsverandering

MRA-services zijn sterk afhankelijk van verschillende HTTPS-verbindingen via de snelwegen naar de CUCM / IM&P / Unity-servers om correct te authenticeren en te verzamelen op de juiste informatie specifiek voor de client die inlogt. Deze communicatie vindt meestal plaats via poorten 8443 en 6972.

Versies onder dan X14.2.0

In versies lager dan X14.2.0, verifieerde de verkeersserver op Expressway-C die die beveiligde HTTPS-verbindingen verwerkt niet het certificaat dat door het verre eind werd voorgelegd. Dit kan leiden tot man-in-the-middle aanvallen. Op de MRA-configuratie is er een optie voor TLS-certificaatverificatie door de configuratie van de 'TLS verify mode' in 'Aan' als u CUCM / IM&P / Unity servers zou toevoegen onder **Configuration > Unified Communications > Unified CM servers / IM en Presence Service knooppunten / Unity Connection servers**. De configuratieoptie en het relevante informatievak worden als voorbeeld weergegeven, wat aangeeft dat de FQDN of IP in het SAN wordt geverifieerd, evenals de geldigheid van het certificaat en of het is ondertekend door een vertrouwde certificeringsinstantie.



Deze TLS-certificaatcontrole wordt alleen uitgevoerd als de CUCM-/IM&P-/Unity-servers zijn gedetecteerd en niet op het moment dat tijdens de MRA-aanmelding de verschillende servers worden bevraagd. Een eerste nadeel van deze configuratie, is dat het slechts het uitgeversadres verifieert u toevoegt. Het bevestigt niet als het certificaat op de abonneeknooppunten correct is opgezet aangezien het de informatie van de abonneeknooppunt (FQDN of IP) van het gegevensbestand van de uitgeversknooppunt terugwint. Een tweede nadeel van deze configuratie is dat wat wordt geadverteerd naar de MRA-klanten als de verbindingsinformatie kan verschillen van het adres van de uitgever die is geplaatst in de Expressway-C-configuratie. Bijvoorbeeld op CUCM, onder **System > Server** kunt u de server met een IP-adres (10.48.36.215 bijvoorbeeld) adverteren en dit wordt dan gebruikt door de MRA-clients (via de Proxied Expressway-verbinding), maar u kunt in de CUCM op Expressway-C toevoegen met de FQDN van cucm.steven.lab. Ga er dus van uit dat het tomcat-certificaat van CUCM cucm.steven.lab als SAN-vermelding bevat maar niet het IP-adres, dan slaagt de ontdekking met 'TLS verify mode' ingesteld op 'On', maar de

werkelijke communicatie van de MRA-clients kan een andere FQDN of IP richten en dus de TLS-verificatie niet doorstaan.

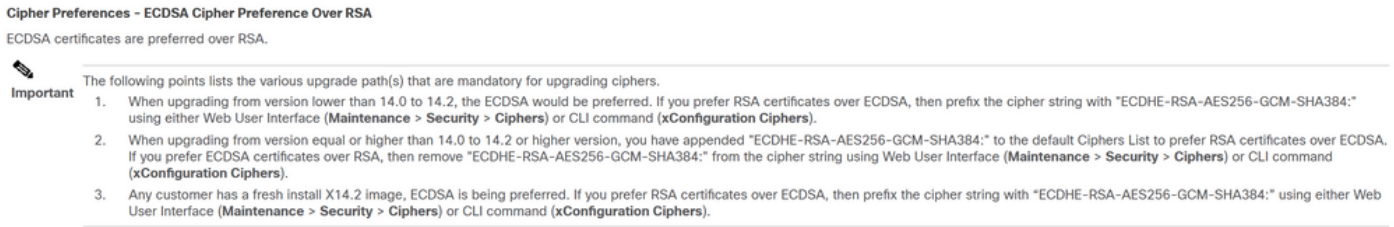
Versies van X14.2.0 en hoger

Vanaf versie X14.2.0 voert de Expressway-server op de TLS-certificaatverificatie uit voor elke HTTPS-aanvraag die via de verkeersserver wordt ingediend. Dat betekent dat dit ook gebeurt wanneer de 'TLS verify mode' is ingesteld op 'Off' tijdens de detectie van de CUCM / IM&P / Unity knooppunten. Wanneer de verificatie niet slaagt, wordt de TLS-handdruk niet voltooid en het verzoek mislukt, wat kan leiden tot verlies van functionaliteit, zoals redundantie of failover-problemen of volledige inlogfouten. Ook met 'TLS verify mode' ingesteld op 'On', is het niet gegarandeerd dat alle verbindingen prima werken zoals later in het voorbeeld wordt besproken.

De exacte certificaten die de snelweg controleert naar de CUCM / IM & P / Unity knooppunten zijn zoals weergegeven in de sectie van de MRA gids.

Naast de standaard TLS-verificatie is er ook een wijziging geïntroduceerd in X14.2 die een andere voorkeursvolgorde zou kunnen adverteren voor de coderingslijst, die afhankelijk is van uw upgradepad. Dit kan leiden tot onverwachte TLS-verbindingen na een software-upgrade, omdat het voor de upgrade kan gebeuren dat het voor het Cisco Tomcat- of Cisco CallManager-certificaat van CUCM (of een ander product met een afzonderlijk certificaat voor ECDSA-algoritme) heeft aangevraagd, maar na de upgrade om de ECDSA-variant verzoekt (wat in feite de veiligere algoritme is dan RSA). De Cisco Tomcat-ECDSA- of Cisco CallManager-ECDSA-certificaten kunnen door een andere CA of alleen maar zelf-ondertekende certificaten worden ondertekend (het standaard).

Deze wijziging van de voorkeursvolgorde van het algoritme is niet altijd relevant voor u, aangezien het afhankelijk is van het upgradepad zoals wordt getoond in de opmerkingen van de release van Expressway X14.2.1. In het kort kunt u van **Onderhoud > Veiligheid > Cijfers** zien voor elk van de cipherlists of het prepend "ECDHE-RSA-AES256-GCM-SHA384:" of niet. Als dit niet het geval is, prefereert hij het nieuwere ECDSA-algoritme boven het RSA-algoritme. Als het wel zo is, dan heb je het gedrag zoals bij RSA dat de hogere voorkeur heeft dan.



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**).

Er zijn twee manieren waarop de TLS-verificatie in dit scenario zou kunnen mislukken, die later in detail worden besproken:

1. CA die het certificaat op afstand heeft ondertekend, wordt niet vertrouwd

a. Zelfondertekend certificaat

b. Certificaat ondertekend door onbekend CA

2. Verbindingsadres (FQDN of IP) is niet in het certificaat opgenomen

Scenario's voor probleemoplossing

De volgende scenario's tonen een gelijkaardig scenario in een laboratoriummilieu waar MRA login na een verbetering van Expressway van X14.0.7 aan X14.2 ontbrak. Zij delen gelijkenissen in de logboeken, nochtans is de resolutie verschillend. De logbestanden worden net verzameld door de diagnostische logboekregistratie (van **Onderhoud > Diagnostiek > Diagnostische logboekregistratie**) die is gestart vóór de MRA-login en is gestopt nadat de MRA-aanmelding is mislukt. Geen extra debug-logboekregistratie is hiervoor ingeschakeld.

1. CA die het certificaat op afstand heeft ondertekend, is niet betrouwbaar

Het certificaat op afstand kan worden ondertekend door een CA die niet is opgenomen in het vertrouwensarchief van de Expressway-C of kan een zelfondertekend certificaat zijn (in essentie ook een CA) dat niet wordt toegevoegd in het vertrouwensarchief van de Expressway-C server.

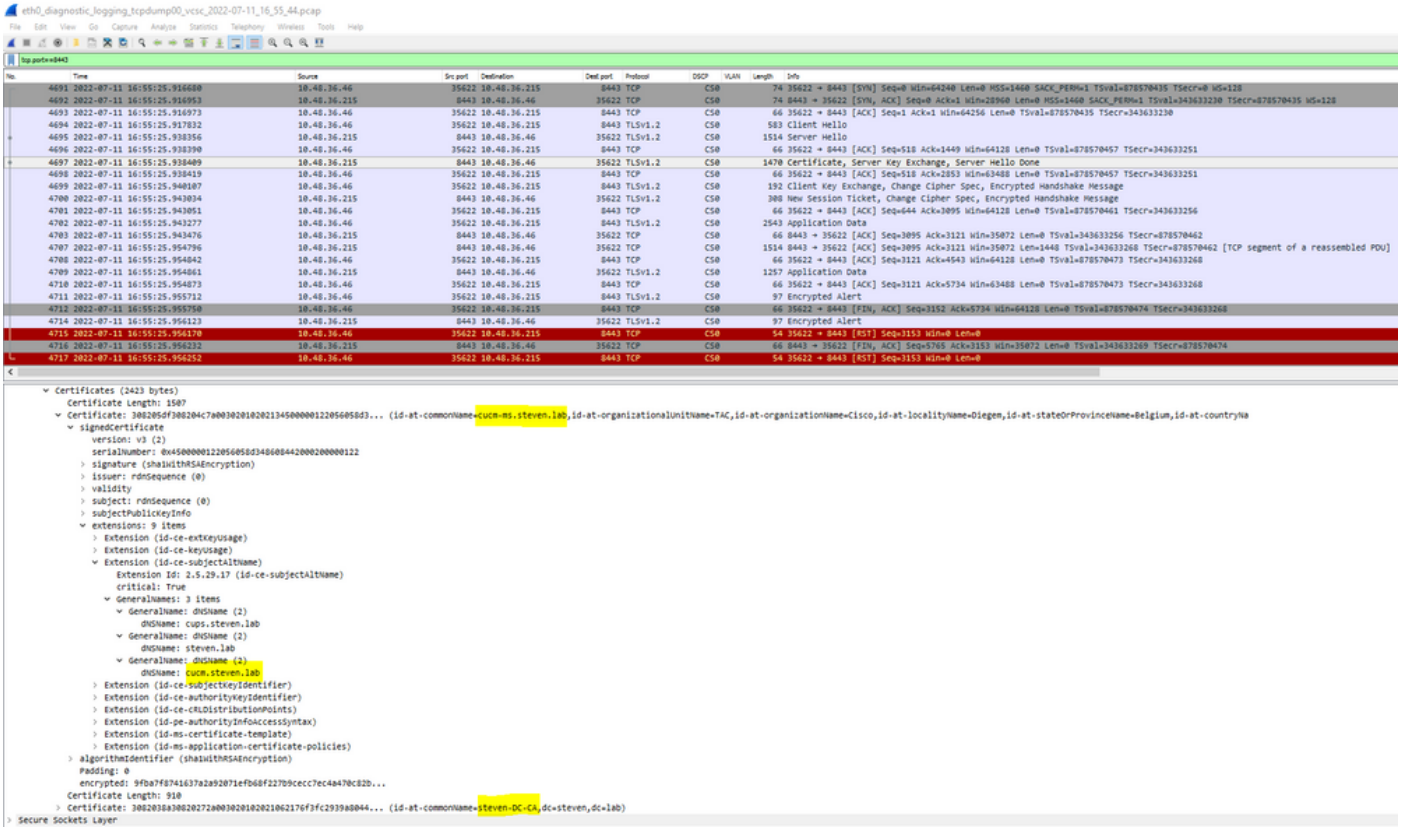
In het voorbeeld hier, kunt u opmerken dat de verzoeken die naar CUCM (10.48.36.215 - cucm.steven.lab) gaan correct worden behandeld op poort 8443 (200 OK reactie) maar het werpt een fout (502 reactie) op poort 6972 voor de TFTP verbinding.

```
===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/c3RldmVuLmxhYi9odHRwcy9jdWNtLnN0ZXZlbi5sYWIvODQ0Mw/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/c3RldmVuLmxhYi9odHRwcy9jdWNtLnN0ZXZlbi5sYWIvNjk3Mg/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 vcsc 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
```

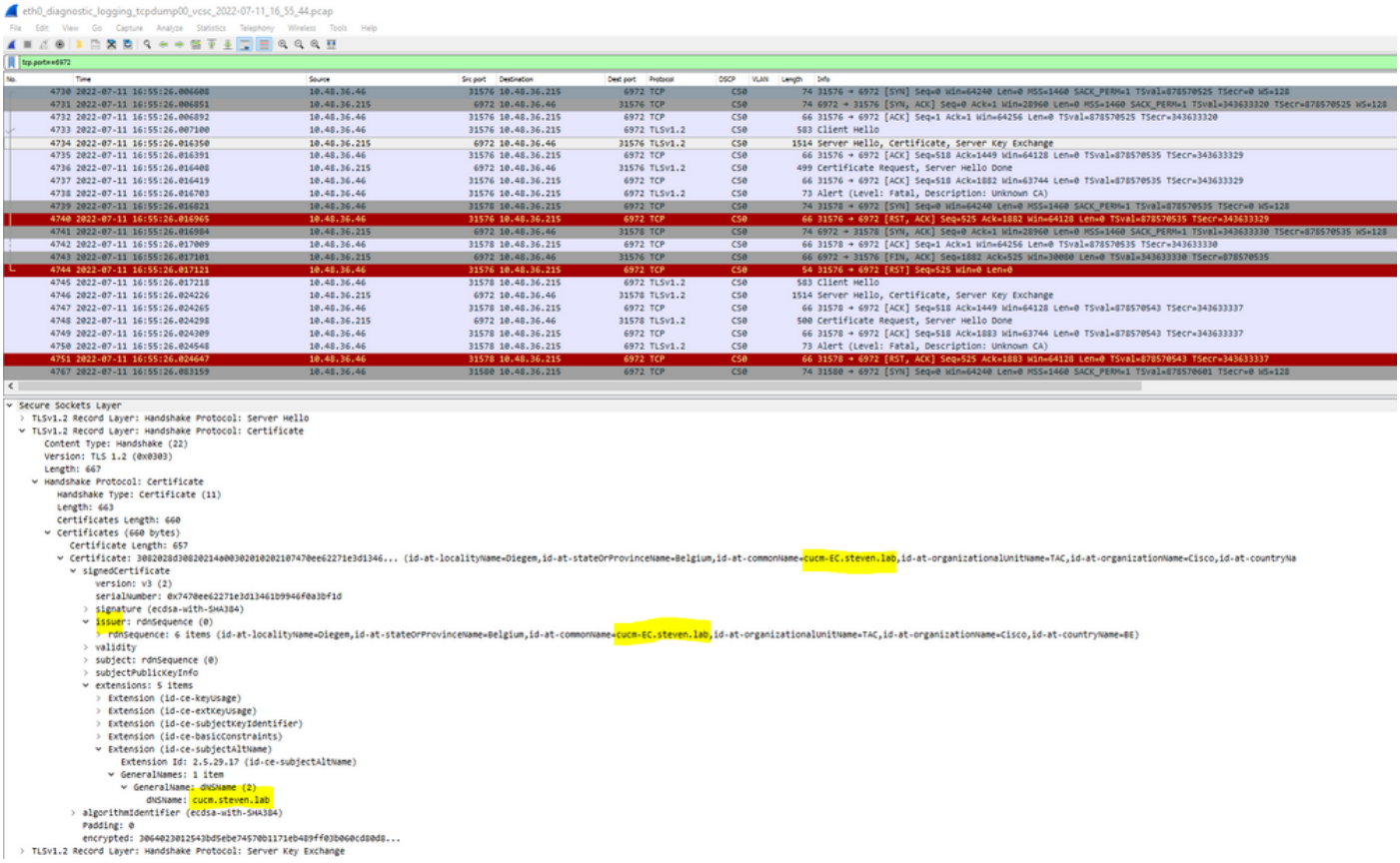
/CSFemusk.cnf.xml HTTP/1.1"
2022-07-11T18:55:26.016+02:00 vcsc traffic_server[18242]: [ET_NET 0] **WARNING: Core server certificate verification failed for** (cucm.steven.lab). **Action=Terminate Error=self signed certificate server=cucm.steven.lab(10.48.36.215) depth=0**
2022-07-11T18:55:26.016+02:00 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 vcsc 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**"

De fout van 'certificate verify faillissement' geeft aan dat Expressway-C de TLS handshake niet kan valideren. De reden voor het, wordt getoond op de waarschuwingslijn aangezien het op een zelf ondertekend certificaat wijst. Als de diepte wordt weergegeven als 0, is het een zelfondertekend certificaat. Wanneer de diepte hoger is dan 0, betekent dit dat het een certificaatketting heeft en dus wordt ondertekend door een onbekend CA (vanuit het perspectief van Expressway-C).

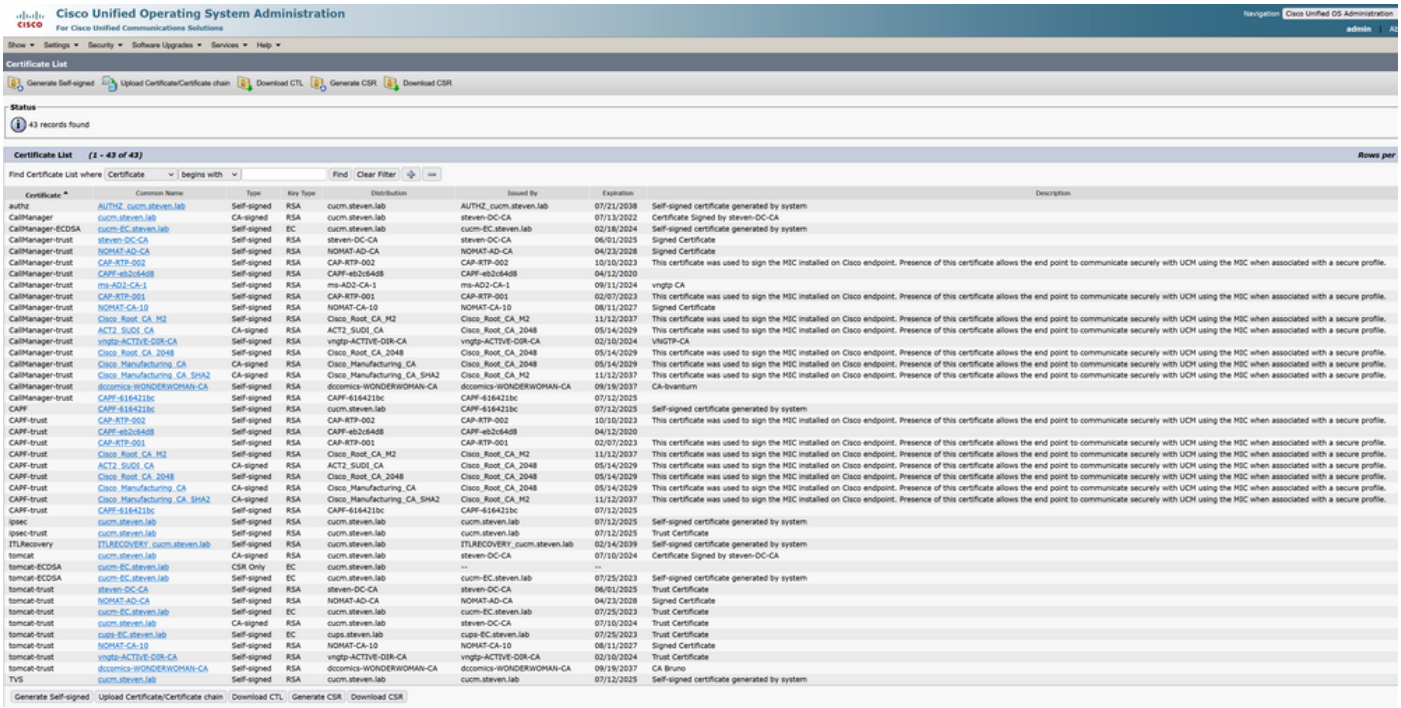
Wanneer we kijken in het pcap-bestand dat is verzameld op de tijdstempels vermeld uit de tekstlogs, kunt u zien dat CUCM het certificaat presenteert met CN als cucm-ms.steven.lab (en cucm.steven.lab als SAN) ondertekend door steven-DC-CA aan de Expressway-C op poort 8443.



Maar als we het certificaat dat op poort 6972 wordt gepresenteerd, inspecteren, kunt u zien dat het een zelfondertekend certificaat is (Issuer is zelf) met CN opgezet als cucm-EC.steven.lab. In de -EC-extensie wordt aangegeven dat dit het ECDSA-certificaat is dat op CUCM is opgesteld.



Op CUCM onder Cisco Unified OS-beheer kunt u de certificaten bekijken die zijn geïnstalleerd onder Security > Certificate Management zoals hier bijvoorbeeld wordt getoond. Het toont een ander certificaat voor tomcat en tomcat-ECDSA waar de tomcat CA ondertekend is (en vertrouwd door de Expressway-C) terwijl het tomcat-ECDSA certificaat zelf ondertekend is en niet vertrouwd door de Expressway-C hier.



2. Verbindingsadres (FQDN of IP) zit niet in het certificaat

Afgezien van de trust store, verifieert er traffic server ook het verbindingsadres waar de MRA client naar vraagt. Bijvoorbeeld, wanneer u hebt ingesteld op CUCM onder **Systeem > Server** uw

CUCM met het IP-adres (10.48.36.215), dan adverteert Expressway-C dit als zodanig aan de client en latere verzoeken van de klant (benaderd via de Expressway-C) zijn gericht op dit adres.

Wanneer dat bepaalde verbindingsadres niet in het servercertificaat is opgenomen, faalt ook de TLS-verificatie en wordt een 502-fout gegenereerd die bijvoorbeeld leidt tot MRA-inlogfout.

```
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
```

Waar c3RldmVuLmxhYi9odHRwcy8xMC40OC4zNi4yMTUvODQ0Mw vertaalt (base64 - https://www.base64decode.org/) naar steven.lab/https/10.48.36.215/8443, waaruit blijkt dat het de verbinding moet maken naar 10.48.36.215 als het aansluitadres in plaats van naar cucm.steven.lab. Zoals in het pakket wordt getoond, bevat het CUCM-tomatencertificaat niet het IP-adres in het SAN en wordt de fout dus gegenereerd.

Eenvoudig valideren

U kunt met de volgende stappen controleren of u gemakkelijk in deze gedragsverandering terechtkomt:

1. Start diagnostische vastlegging op Expressway-E- en C-server(s) (idealiter met TCPDumps ingeschakeld) vanaf **Onderhoud > Diagnostiek > Diagnostische vastlegging** (in het geval van een cluster is het voldoende om deze te starten vanaf het primaire knooppunt)

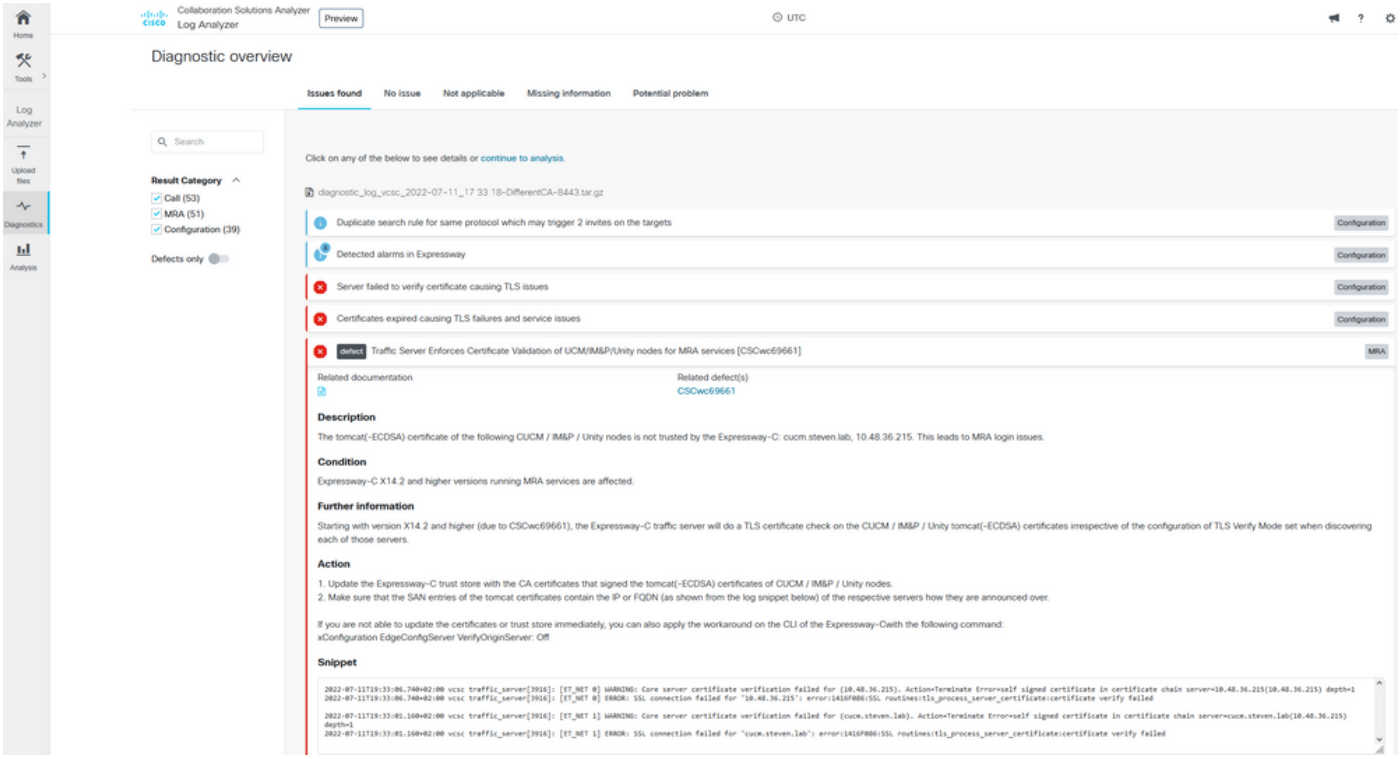
2. Probeer een MRA-login of test de defecte functionaliteit na de upgrade

3. Wacht tot het mislukt en stop vervolgens met de diagnostische vastlegging op Expressway-E- en C-server(s) (zorg er in het geval van een cluster voor dat u de logbestanden van elk knooppunt van het cluster afzonderlijk verzamelt)

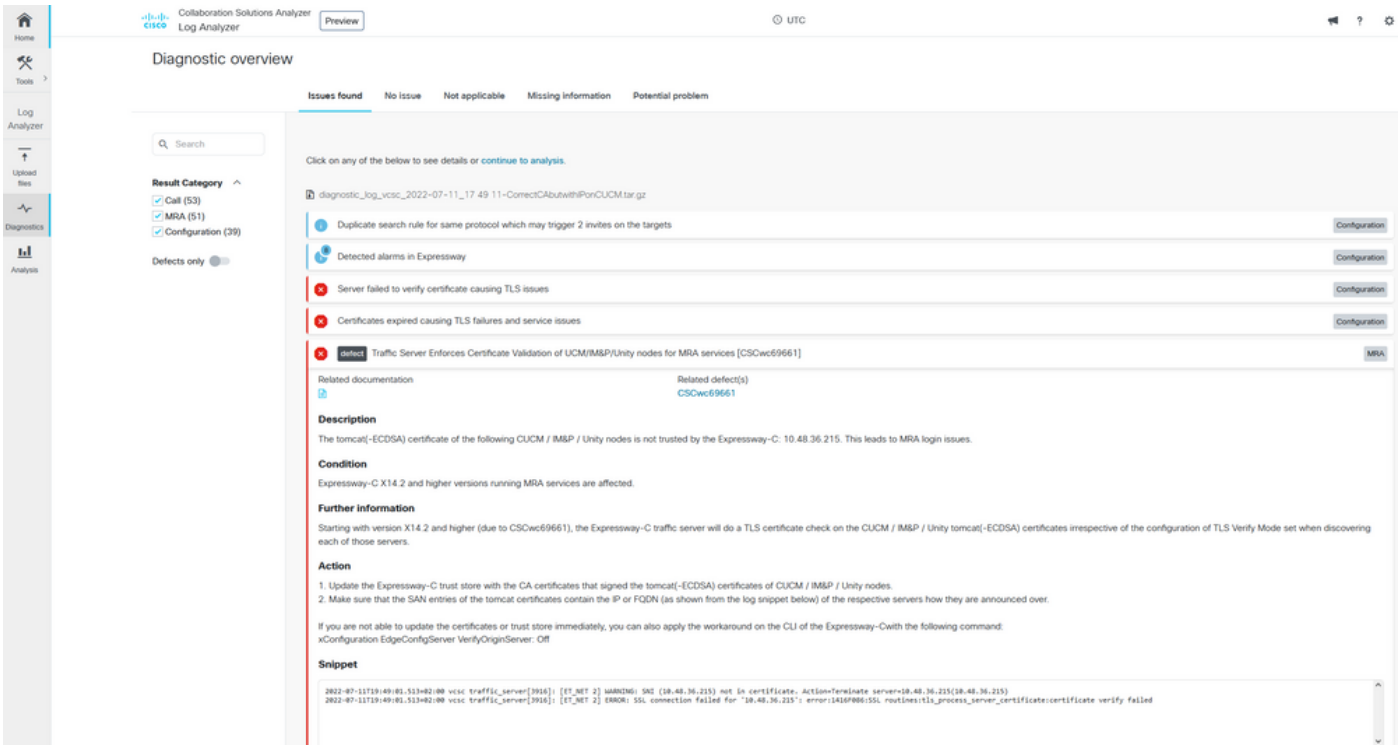
4. Upload en analyseer de logs in de Collaboration Solution Analyzer-tool

5. Als u het probleem tegenkomt, worden de meest recente waarschuwings- en foutregels met

betrekking tot deze wijziging voor elk van de betrokken servers opgenomen



CA diagnostische handtekening



SNI diagnostische handtekening

Oplossing

De langetermijnoplossing is ervoor te zorgen dat de TLS-verificatie prima verloopt. Welke actie u moet uitvoeren, is afhankelijk van het weergegeven waarschuwingsbericht.

Wanneer u de *WAARSCHUWING* bekijkt*: Verificatie van basisservercertificaat mislukt voor (<server-FQDN-or-IP>). Action=Terminate Error=self-signed certificate server=cucm.steven.lab(10.48.36.215)* ***deep=x*** bericht, dan moet u de vertrouwensopslag op de

Expressway-C servers dienovereenkomstig bijwerken. Ofwel met de CA-keten die dit certificaat ondertekende (diepte > 0) of met het zelfondertekende certificaat (diepte = 0) van **Onderhoud > Beveiliging > Betrouwbaar CA-certificaat**. Zorg ervoor dat u deze actie uitvoert op elke server in het cluster. Een andere optie zou zijn om het certificaat op afstand te ondertekenen door een bekende CA in de Expressway-C trust store.

> **Opmerking: Expressway staat niet toe om twee verschillende (zelf-ondertekende bijvoorbeeld) certificaten te uploaden in de vertrouwenswinkel van Expressway die dezelfde algemene naam (CN) hebben als volgens Cisco bug ID CSCwa12905. Om dit te corrigeren, ga naar CA-ondertekende certificaten of upgrade uw CUCM naar versie 14 waar u hetzelfde (zelf-ondertekende) certificaat voor Tomcat en CallManager kunt hergebruiken.**

Wanneer u de *WAARSCHUWING* bekijkt*: SNI (<server-FQDN-or-IP>) niet in* certificaatbericht, dan geeft het aan dat deze server FQDN of IP niet is opgenomen in het certificaat dat werd gepresenteerd. Ofwel kunt u het certificaat aanpassen om die informatie op te nemen, ofwel kunt u de configuratie wijzigen (zoals op CUCM op System > Server) om iets in het servercertificaat op te nemen en vervolgens de configuratie op de Expressway-C server te ververversen om er rekening mee te houden.

De kortetermijnoplossing is om de tijdelijke oplossing toe te passen zoals gedocumenteerd om terug te vallen op het vorige gedrag vóór X14.2.0. U kunt op dit gebied presteren via de CLI op de Expressway-C serverknooppunten met de nieuw geïntroduceerde opdracht:

```
xConfiguration EdgeConfigServer VerifyOriginServer: Off
```

Over deze vertaling

Cisco heeft dit document vertaald via een combinatie van machine- en menselijke technologie om onze gebruikers wereldwijd ondersteuningscontent te bieden in hun eigen taal. Houd er rekening mee dat zelfs de beste machinevertaling niet net zo nauwkeurig is als die van een professionele vertaler. Cisco Systems, Inc. is niet aansprakelijk voor de nauwkeurigheid van deze vertalingen en raadt aan altijd het oorspronkelijke Engelstalige document (link) te raadplegen.