Dépannage de la vérification du certificat du serveur de trafic Expressway pour les services MRA introduits par CSCwc69661 / CSCwa25108

Table des matières

Introduction

Ce document décrit le changement de comportement sur les versions Expressway de X14.2.0 et ultérieures liées à l'ID de bogue Cisco CSCwc69661 ou à l'ID de bogue Cisco CSCwa25108.

Conditions préalables

Exigences

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Configuration de base d'Expressway
- Configuration de base MRA

Composants utilisés

Les informations contenues dans ce document sont basées sur Cisco Expressway version X14.2 et ultérieures.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau

est en ligne, assurez-vous de bien comprendre l’incidence possible des commandes.

Informations générales

Avec ce changement de comportement marqué par l'ID de bogue Cisco CSCwc69661 ou ID de bogue Cisco CSCwa25108 , le serveur de trafic sur la plate-forme Expressway effectue la vérification de certificat des noeuds de serveur Cisco Unified Communication Manager (CUCM), Cisco Unified Instant Messaging & Presence (IM&P) et Unity pour les services Mobile and Remote Access (MRA). Cette modification peut entraîner des échecs de connexion MRA après une mise à niveau sur votre plateforme Expressway.

Le protocole HTTPS (Hypertext Transfer Protocol Secure) est un protocole de communication sécurisé qui utilise le protocole TLS (Transport Layer Security) pour chiffrer la communication. Il crée ce canal sécurisé en utilisant un certificat TLS qui est échangé dans la connexion TLS. De cette façon, il sert deux objectifs : l'authentification (pour savoir à qui vous vous connectez) et la confidentialité (le chiffrement). L'authentification protège contre les attaques de l'homme du milieu et la confidentialité empêche les pirates d'écouter et de falsifier la communication.

La vérification TLS (certificat) est effectuée en vue de l'authentification et vous permet de vous assurer que vous êtes connecté à la partie distante appropriée. La vérification se compose de deux éléments individuels :

1. Chaîne d’autorités de certification (AC) de confiance

2. Autre nom du sujet (SAN) ou nom commun (CN)

Chaîne CA approuvée

Pour qu’Expressway-C puisse faire confiance au certificat que CUCM / IM&P / Unity envoie, il doit être en mesure d’établir un lien entre ce certificat et une autorité de certification (CA) de niveau supérieur (racine) à laquelle il fait confiance. Un tel lien, une hiérarchie de certificats qui lie un certificat d'entité à un certificat d'autorité de certification racine, est appelé une chaîne de confiance. Pour pouvoir vérifier une telle chaîne de confiance, chaque certificat contient deux champs : Émetteur (ou 'Émis par') et Objet (ou 'Émis à').

Les certificats de serveur, tels que celui que CUCM envoie à Expressway-C, ont généralement leur nom de domaine complet (FQDN) dans le champ « Subject » du CN :

```
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
```

Exemple de certificat de serveur pour CUCM cucm.vngtp.lab. Il contient le nom de domaine complet dans l'attribut CN du champ Objet ainsi que d'autres attributs tels que le pays (C), l'état (ST), l'emplacement (L), ... Nous pouvons également voir que le certificat du serveur est distribué (émis) par une autorité de certification appelée vngtp-ACTIVE-DIR-CA.

Les autorités de certification de niveau supérieur (CA racine) peuvent également émettre un certificat pour s’identifier. Dans ce certificat d'autorité de certification racine, nous voyons que l'émetteur et l'objet ont la même valeur :

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

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

Il s'agit d'un certificat distribué par une autorité de certification racine pour s'identifier.

Dans une situation typique, les autorités de certification racine n'émettent pas directement de certificats de serveur. Au lieu de cela, ils émettent des certificats pour d'autres CA. Ces autres AC sont alors appelées AC intermédiaires. Les autorités de certification intermédiaires peuvent à leur tour émettre directement des certificats de serveur ou des certificats pour d'autres autorités de certification intermédiaires. Nous pouvons avoir une situation où un certificat de serveur est émis par l'intermédiaire CA 1, qui à son tour obtient un certificat de l'intermédiaire CA 2 et ainsi de suite. Jusqu'à ce que l'autorité de certification intermédiaire obtienne son certificat directement de l'autorité de certification racine :

```
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
```

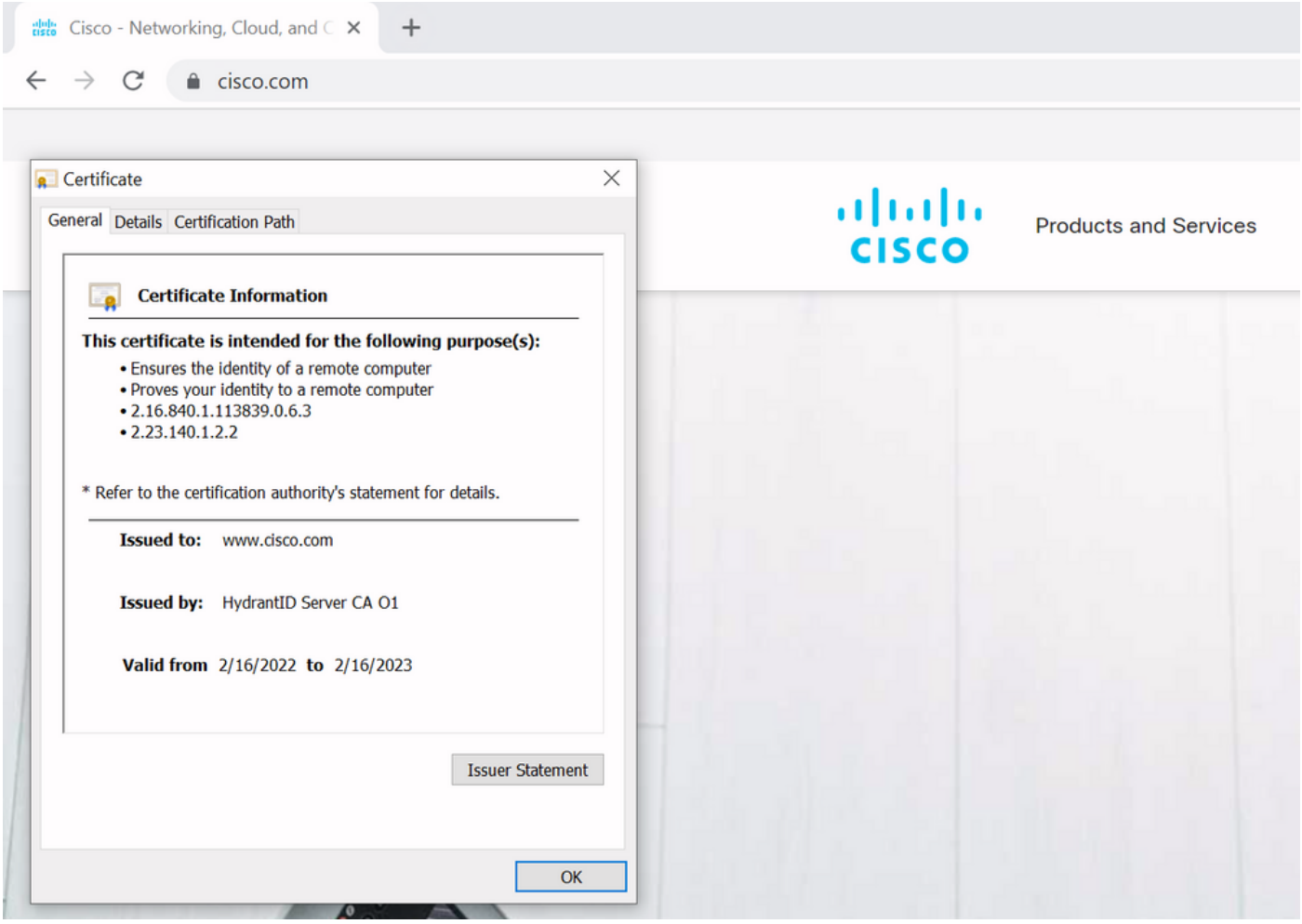
Maintenant, pour qu'Expressway-C puisse faire confiance au certificat de serveur que CUCM envoie, il doit être en mesure de construire la chaîne de confiance à partir de ce certificat de serveur jusqu'à un certificat d'autorité de certification racine. Pour ce faire, nous devons télécharger le certificat d'autorité de certification racine ainsi que tous les certificats d'autorité de certification intermédiaires (s'il y en a, ce qui n'est pas le cas si l'autorité de certification racine aurait directement émis le certificat de serveur de CUCM) dans le magasin de confiance d'Expressway-C.

> **Note: Bien que les champs Émetteur et Objet soient faciles à créer dans la chaîne de confiance d'une manière lisible, CUCM n'utilise pas ces champs dans le certificat. Il utilise plutôt les champs « Identificateur de clé d'autorité X509v3 » et « Identificateur de clé d'objet X509v3 » pour créer la chaîne de confiance. Ces clés contiennent des identifiants pour les certificats qui sont plus précis que pour utiliser les champs Subject/Issuer : il peut y avoir 2 certificats avec les mêmes champs Objet/Émetteur, mais l'un d'eux a expiré et l'autre est toujours valide. Ils auraient tous deux un identifiant de clé d'objet X509v3 différent, de sorte que CUCM puisse toujours déterminer la chaîne de confiance correcte.**
>
> **Ce n'est pas le cas pour Expressway, bien que selon l'ID de bogue Cisco CSCwa12905 et qu'il ne soit pas possible de télécharger deux certificats différents (auto-signés par exemple) dans le magasin de confiance d'Expressway qui ont le même nom commun (CN). La façon de corriger cela, c'est d'utiliser des certificats signés par l'autorité de certification ou d'utiliser des noms communs différents pour cela ou de voir qu'il utilise toujours le même certificat (potentiellement via la fonctionnalité de réutilisation de certificat dans CUCM 14).**

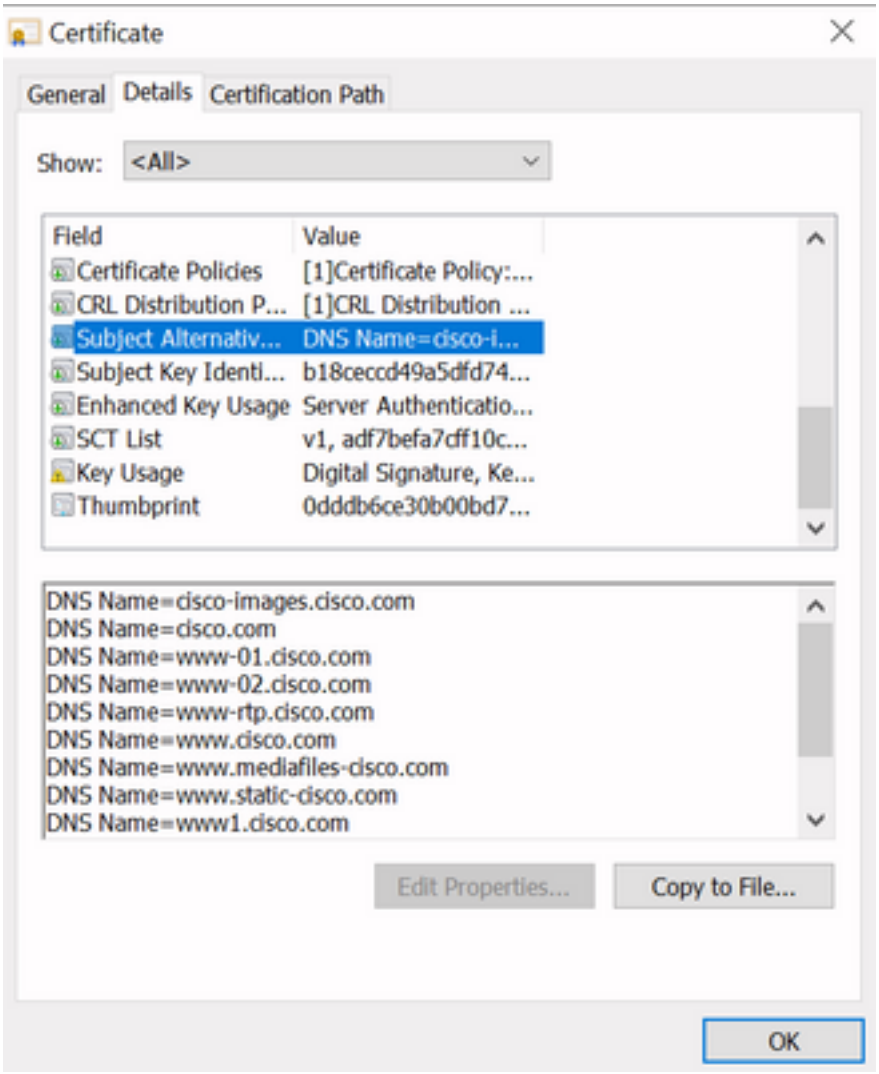
Vérification SAN ou CN

L'étape 1 vérifie le magasin d'approbations, mais toute personne qui a un certificat signé par une autorité de certification dans le magasin d'approbations serait alors valide. Cela n'est évidemment pas suffisant. Par conséquent, il y a une vérification supplémentaire qui confirme que le serveur auquel vous vous connectez spécifiquement est bien le bon. Il le fait en fonction de l'adresse pour laquelle la demande a été faite.

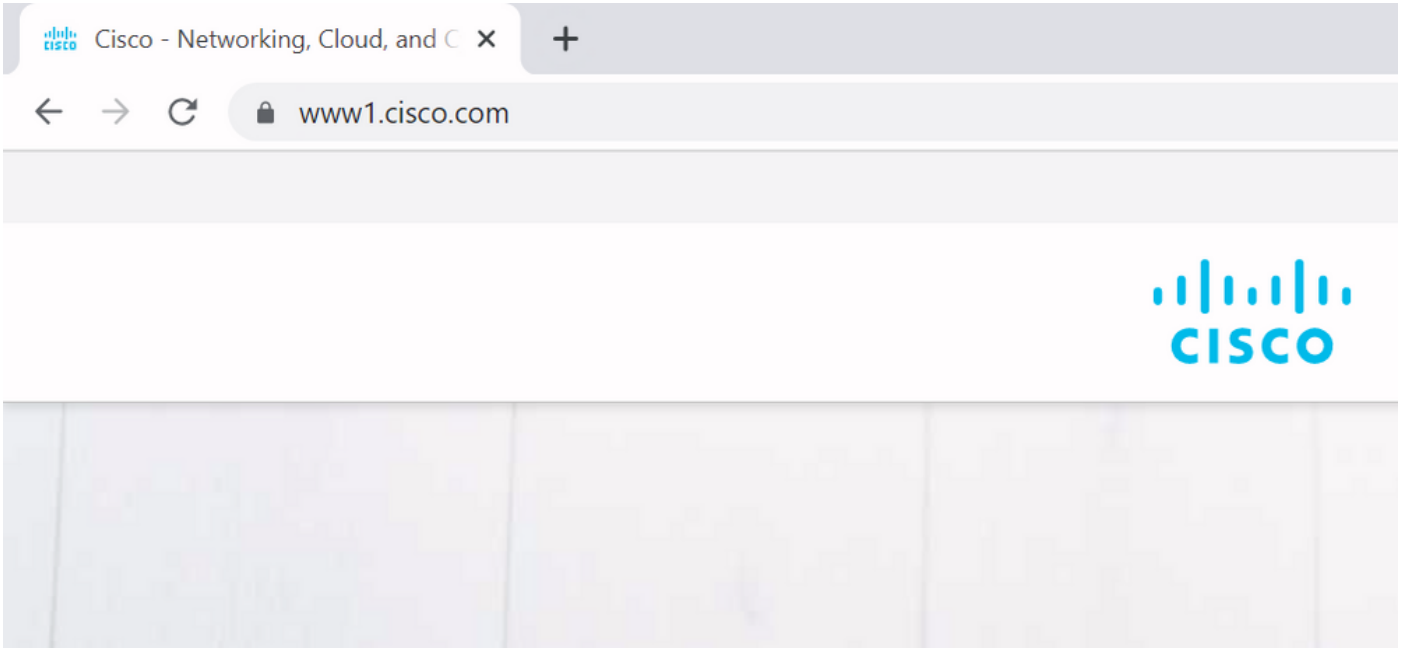
Le même type d'opération se produit dans votre navigateur, alors laissez-nous regarder à travers un exemple. Si vous naviguez vers https://www.cisco.com, vous voyez une icône de verrouillage à côté de l'URL que vous avez entrée et cela signifie qu'il s'agit d'une connexion approuvée. Cela est basé à la fois sur la chaîne de confiance CA (de la première section) ainsi que sur le contrôle SAN ou CN. Si nous ouvrons le certificat (via le navigateur par un clic sur l'icône de verrouillage), vous voyez que le nom commun (vu sur le champ 'Émis à : ') est défini sur www.cisco.com et qui correspond exactement à l'adresse à laquelle nous voulions nous connecter. De cette façon, il peut être sûr que nous nous connectons au bon serveur (parce que nous faisons confiance à l'autorité de certification qui a signé le certificat et qui effectue la vérification avant qu'il distribue le certificat).



Lorsque nous examinons les détails du certificat et en particulier les entrées SAN, nous voyons que la même chose est répétée ainsi que d'autres FQDN :

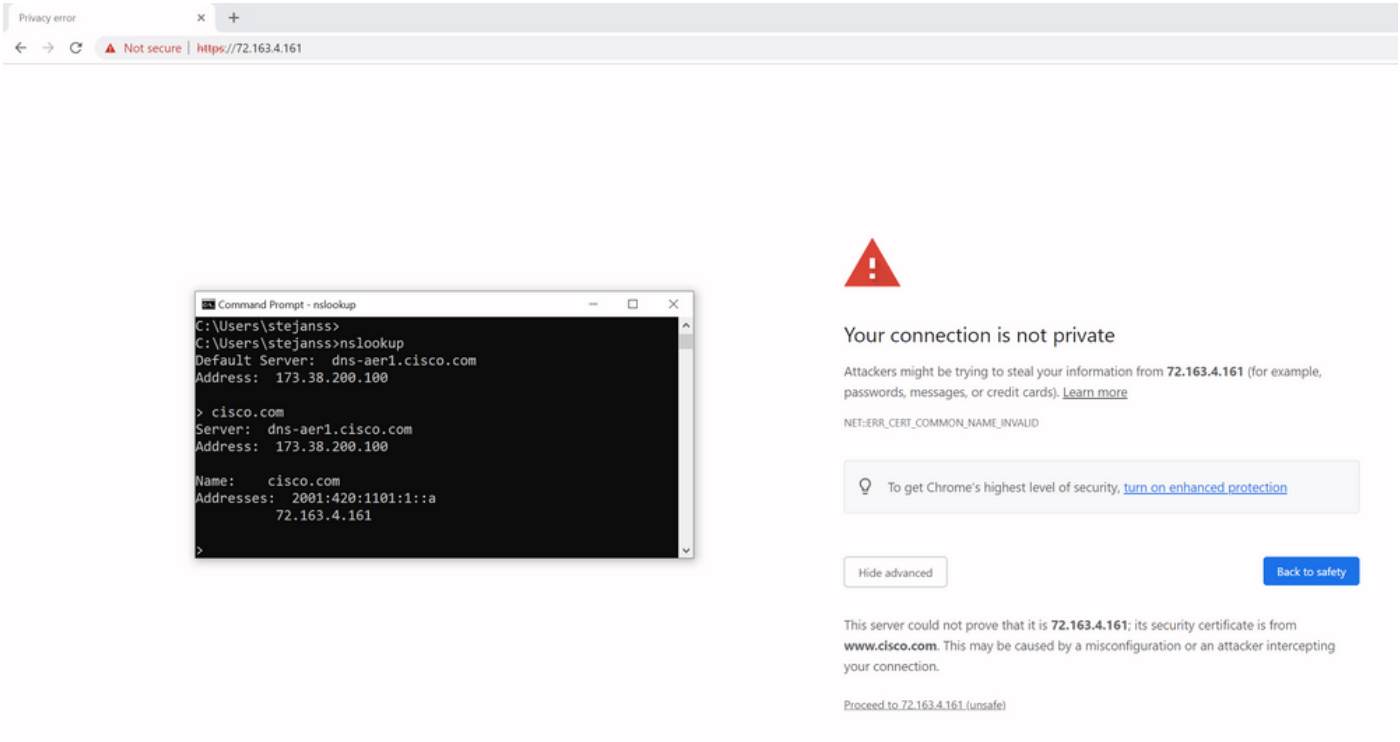


Cela signifie que lorsque nous demandons à nous connecter à https://www1.cisco.com par exemple, cela s'affiche également comme une connexion sécurisée car elle est contenue dans les entrées SAN.



Cependant, lorsque nous ne naviguons pas vers https://www.cisco.com mais plutôt directement vers l'adresse IP (https://72.163.4.161), il n'affiche pas de connexion sécurisée car il fait confiance à l'autorité de certification qui l'a signé mais le certificat qui nous a été présenté ne contient pas

l'adresse (72.163.4.161) que nous avons utilisée pour nous connecter au serveur.



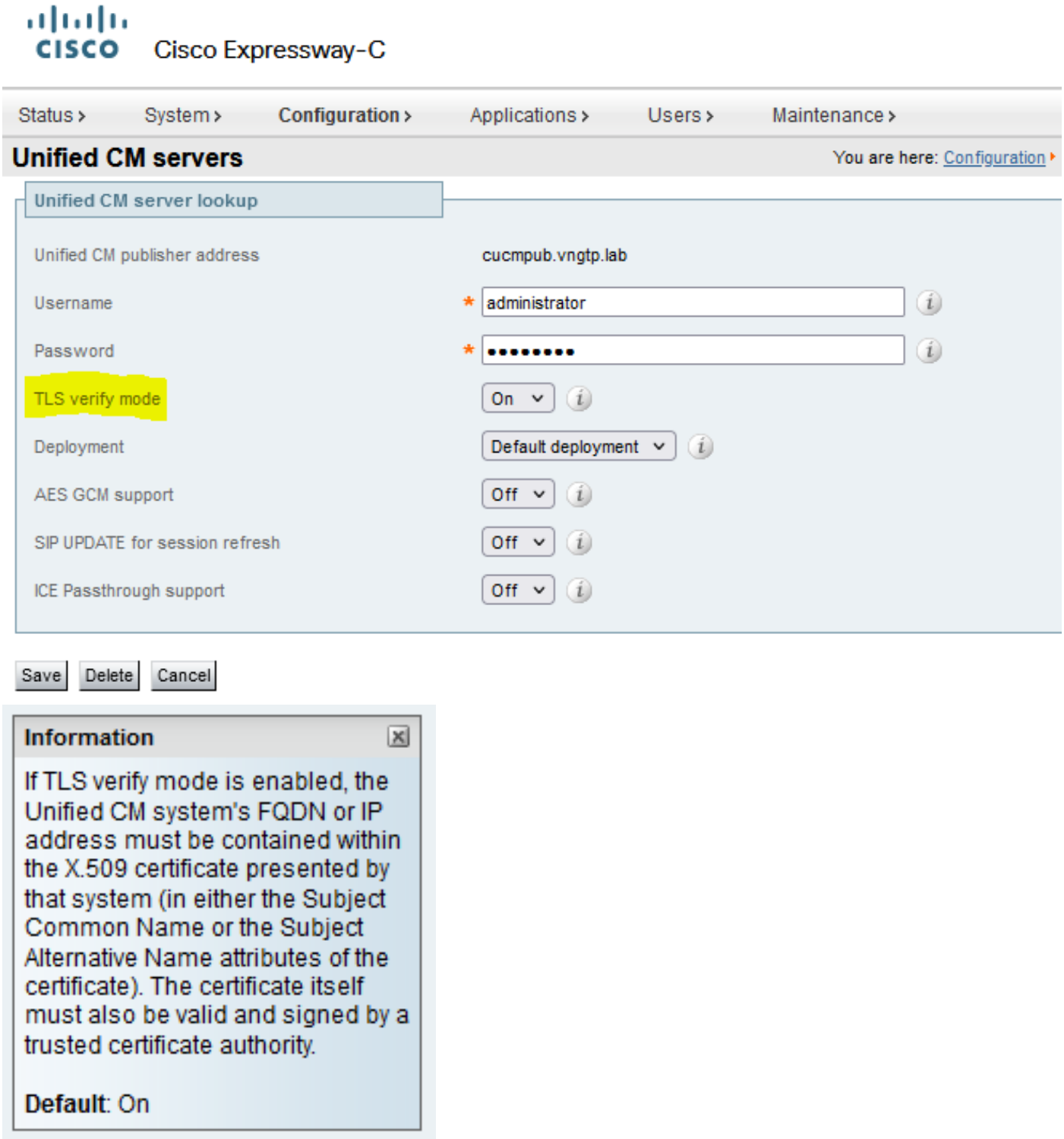
Dans le navigateur, vous pouvez contourner ce paramètre, mais il s'agit d'un paramètre que vous pouvez activer sur les connexions TLS et qui n'est pas autorisé. Par conséquent, il est important que vos certificats contiennent les bons noms CN ou SAN que la partie distante prévoit d'utiliser afin de se connecter.

Changement De Comportement

Les services MRA s'appuient fortement sur plusieurs connexions HTTPS via Expressways vers les serveurs CUCM / IM&P / Unity pour s'authentifier correctement et collecter les bonnes informations spécifiques au client qui se connecte. Cette communication se produit généralement sur les ports 8443 et 6972.

Versions inférieures à X14.2.0

Dans les versions antérieures à X14.2.0, le serveur de trafic sur Expressway-C qui gère ces connexions HTTPS sécurisées n'a pas vérifié le certificat qui a été présenté par l'extrémité distante. Cela pourrait conduire à des attaques de l'homme du milieu. Dans la configuration MRA, il y a une option pour la vérification du certificat TLS par la configuration du 'Mode de vérification TLS' à 'Activé' quand vous ajouteriez soit CUCM / IM&P / serveurs Unity sous **Configuration > Communications unifiées > serveurs Unified CM / noeuds IM and Presence Service / serveurs Unity Connection**. L'option de configuration et la boîte d'informations correspondante sont présentées à titre d'exemple, ce qui indique qu'il vérifie le nom de domaine complet ou l'adresse IP dans le SAN, ainsi que la validité du certificat et s'il est signé par une autorité de certification de confiance.



Cette vérification de certificat TLS n'est effectuée qu'au moment de la découverte des serveurs CUCM / IM&P / Unity et non au moment de la connexion MRA où les différents serveurs sont interrogés. Un premier inconvénient de cette configuration est qu'elle ne vérifie que l'adresse d'éditeur que vous ajoutez. Il ne vérifie pas si le certificat sur les noeuds d'abonné a été correctement configuré lorsqu'il récupère les informations de noeud d'abonné (FQDN ou IP) dans la base de données du noeud éditeur. Un deuxième inconvénient de cette configuration est que ce qui est annoncé aux clients MRA comme informations de connexion peut être différent de l'adresse de l'éditeur qui a été placée dans la configuration d'Expressway-C. Par exemple, sur CUCM, sous **System > Server** vous pouvez annoncer le serveur avec une adresse IP (10.48.36.215 par exemple) et ceci est ensuite utilisé par les clients MRA (via la connexion Expressway proxy) mais vous pouvez ajouter le CUCM sur Expressway-C avec le FQDN de cucm.steven.lab. Supposons donc que le certificat tomcat de CUCM contient cucm.steven.lab comme entrée SAN mais pas l'adresse IP, puis la détection avec 'TLS Verify Mode' défini sur 'On'

réussit mais les communications réelles des clients MRA peuvent cibler un FQDN ou IP différent et donc échouer la vérification TLS.

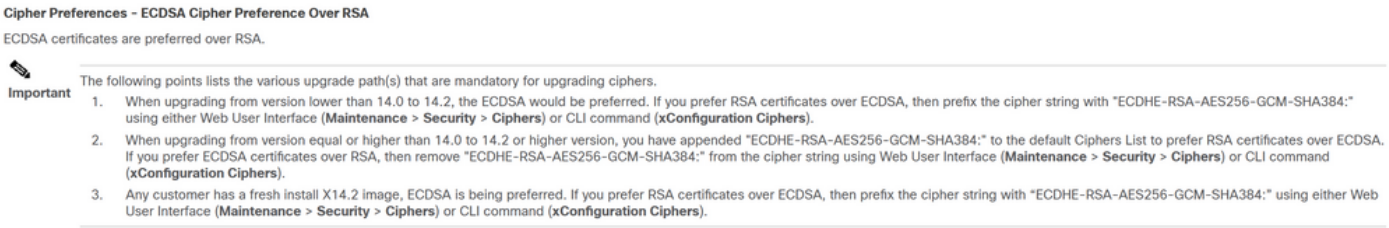
Versions de X14.2.0 et ultérieures

Àpartir de la version X14.2.0, le serveur Expressway effectue la vérification de certificat TLS pour chaque requête HTTPS unique qui est faite par le serveur de trafic. Cela signifie qu'il effectue également cette opération lorsque le mode de vérification TLS est défini sur Désactivé lors de la détection des noeuds CUCM / IM&P / Unity. Lorsque la vérification échoue, la connexion TLS ne se termine pas et la demande échoue, ce qui peut entraîner une perte de fonctionnalité, comme des problèmes de redondance ou de basculement, ou des échecs de connexion complets, par exemple. De même, lorsque le paramètre « TLS Verify Mode » est activé, cela ne garantit pas que toutes les connexions fonctionnent correctement, comme indiqué dans l'exemple ci-après.

Les certificats exacts que l'Expressway vérifie vers les noeuds CUCM / IM&P / Unity sont comme indiqué dans la section du guide MRA.

En plus de la vérification TLS par défaut, il y a aussi un changement introduit dans X14.2 qui pourrait annoncer un ordre de préférence différent pour la liste de chiffrement, qui dépend de votre chemin de mise à niveau. Cela peut provoquer des connexions TLS inattendues après une mise à niveau logicielle, car il peut arriver qu'avant la mise à niveau, il ait demandé le certificat Cisco Tomcat ou Cisco CallManager de CUCM (ou de tout autre produit disposant d'un certificat distinct pour l'algorithme ECDSA), mais qu'après la mise à niveau, il demande la variante ECDSA (qui est la variante de chiffrement plus sécurisée en fait que RSA). Les certificats Cisco Tomcat-ECDSA ou Cisco CallManager-ECDSA peuvent être signés par une autre autorité de certification ou simplement par des certificats auto-signés (par défaut).

Cette modification de l'ordre de préférence de chiffrement ne vous concerne pas toujours, car elle dépend du chemin de mise à niveau indiqué dans les notes de version d'Expressway X14.2.1. En bref, vous pouvez voir de **Maintenance > Security > Ciphers** pour chacune des listes de chiffrement si elle ne précède pas "ECDHE-RSA-AES256-GCM-SHA384:" ou non. Si ce n'est pas le cas, il préfère le nouveau chiffrement ECDSA au chiffrement RSA. Si c'est le cas, vous avez le comportement précédent avec RSA qui a la préférence la plus élevée.


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**).

Dans ce scénario, la vérification TLS peut échouer de deux manières, qui sont décrites en détail plus loin :

1. L'autorité de certification qui a signé le certificat distant n'est pas approuvée

a. Certificat auto-signé

b. Certificat signé par une CA inconnue

2. L'adresse de connexion (FQDN ou IP) ne figure pas dans le certificat

Scénarios de dépannage

Les scénarios suivants présentent un scénario similaire dans un environnement de travaux pratiques où la connexion MRA a échoué après une mise à niveau d'Expressway de X14.0.7 à X14.2. Ils partagent des similitudes dans les journaux, mais la résolution est différente. Les journaux sont simplement collectés par la journalisation de diagnostic (à partir de **Maintenance > Diagnostics > Journalisation de diagnostic**) qui a commencé avant la connexion MRA et qui s'est arrêtée après l'échec de la connexion MRA. Aucune journalisation de débogage supplémentaire n'a été activée pour cette application.

1. L'Autorité De Certification Qui A Signé Le Certificat Distant N'Est Pas Approuvée

Le certificat distant peut soit être signé par une CA qui n'est pas incluse dans le magasin de confiance de l'Expressway-C, soit être un certificat auto-signé (en fait aussi une CA) qui n'est pas ajouté dans le magasin de confiance du serveur de l'Expressway-C.

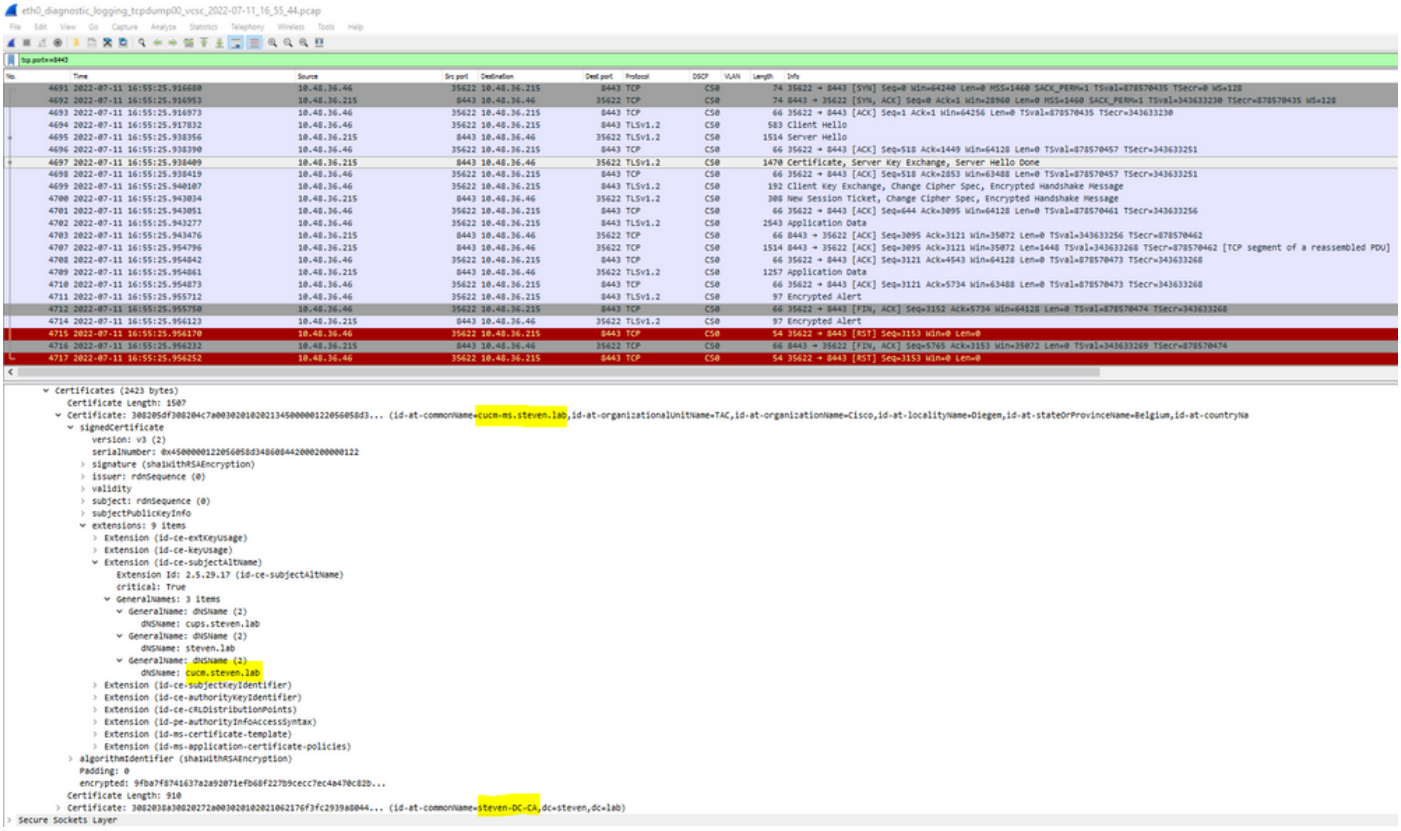
Dans cet exemple, vous pouvez observer que les requêtes qui vont à CUCM (10.48.36.215 - cucm.steven.lab) sont traitées correctement sur le port 8443 (réponse 200 OK) mais cela génère une erreur (réponse 502) sur le port 6972 pour la connexion TFTP.

```
===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
```

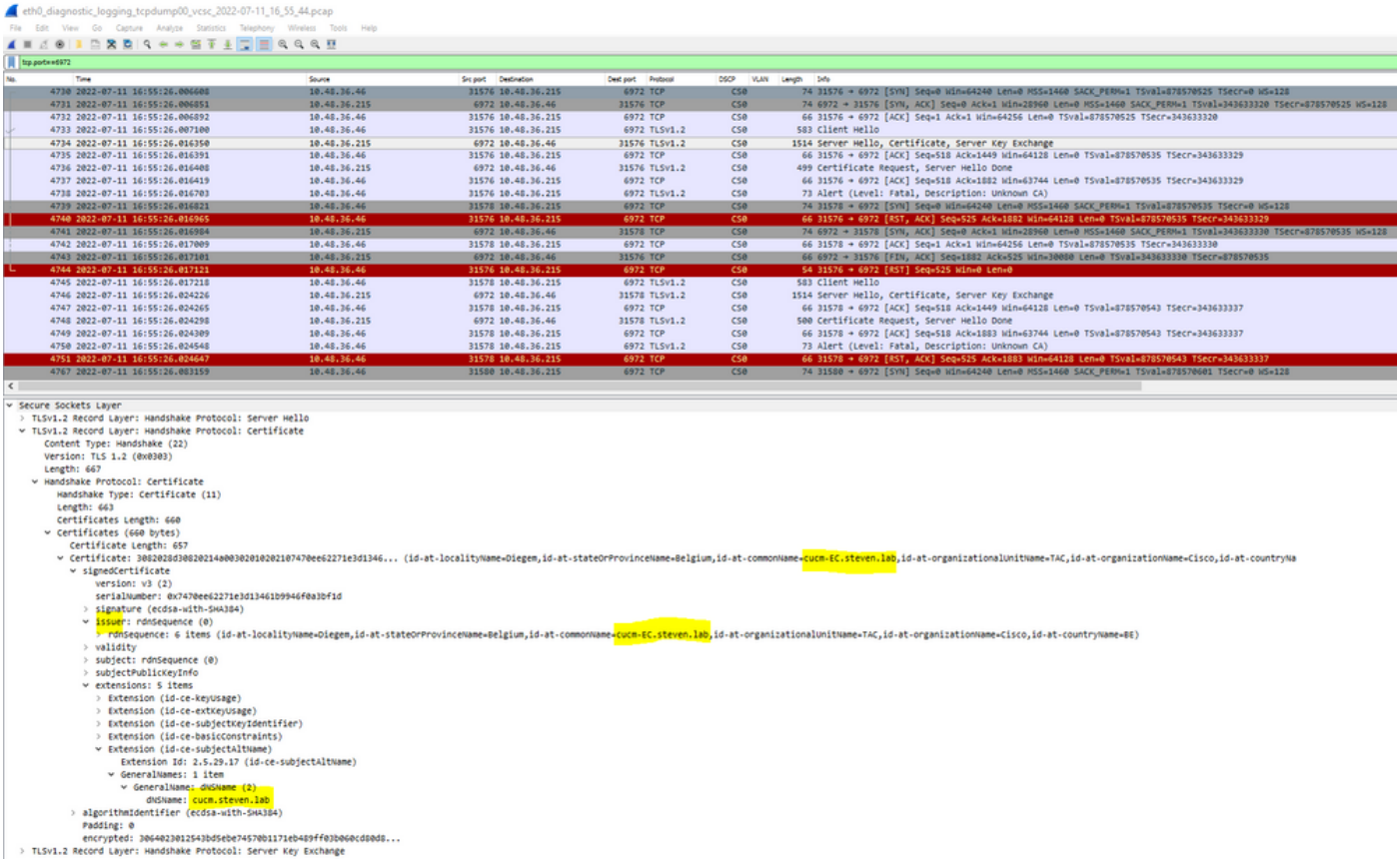
```
/CSFemusk.cnf.xml HTTP/1.1"
2022-07-11T18:55:26.016+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:26,016"
Module="network.http.trafficserver" Level="INFO": Detail="Sending Request" Txn-id="191"
TrackingID="bb0c8492-8c15-4537-a7d1-082dde781dbd" Dst-ip="10.48.36.215" Dst-port="6972" Msg="GET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

L'erreur « certificate verify failed » indique que l'Expressway-C n'a pas pu valider la connexion TLS. La raison de cette erreur est indiquée sur la ligne d'avertissement car elle indique un certificat auto-signé. Si la profondeur est 0, il s'agit d'un certificat auto-signé. Lorsque la profondeur est supérieure à 0, cela signifie qu'il a une chaîne de certificats et donc qu'il est signé par une CA inconnue (du point de vue d'Expressway-C).

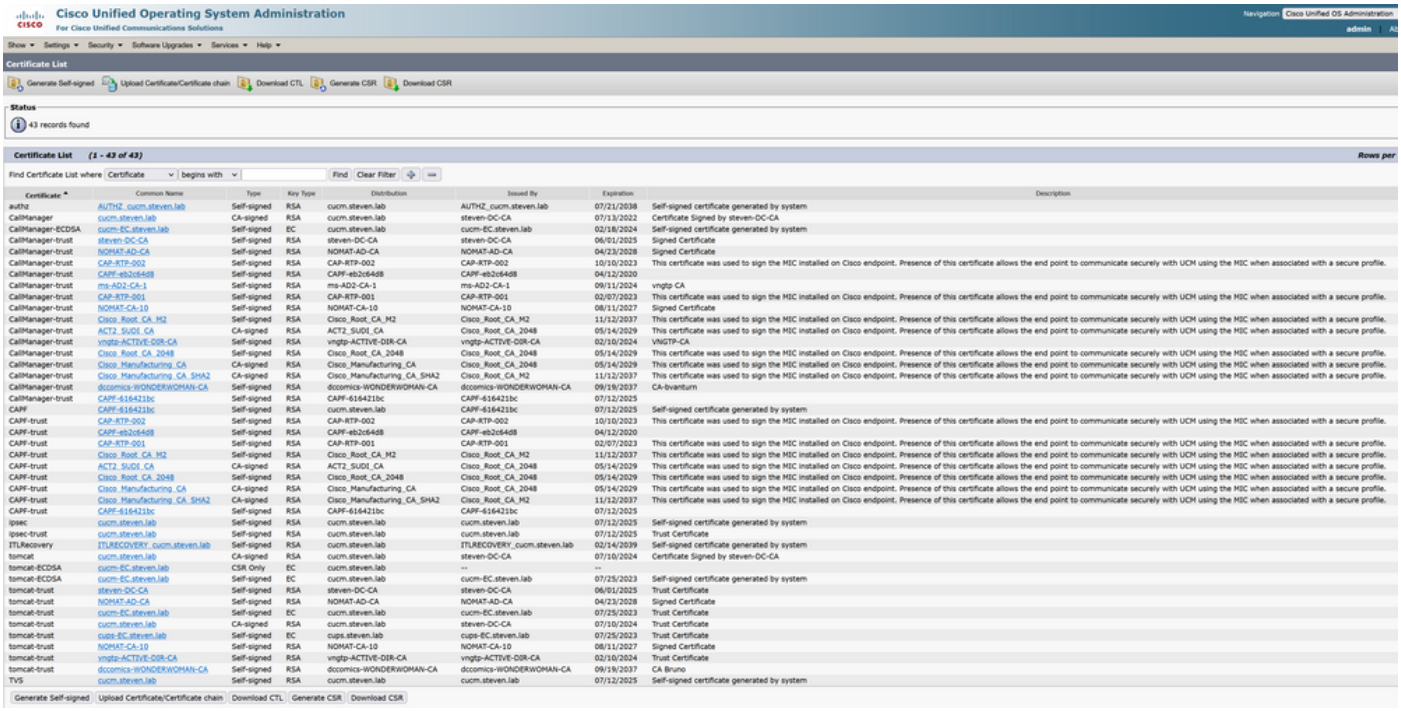
Lorsque nous regardons dans le fichier pcap qui a été collecté aux horodatages mentionnés dans les journaux de texte, vous pouvez voir que CUCM présente le certificat avec CN comme cucm-ms.steven.lab (et cucm.steven.lab comme SAN) signé par steven-DC-CA à l'Expressway-C sur le port 8443.



Mais lorsque nous examinons le certificat présenté sur le port 6972, vous pouvez voir qu'il s'agit d'un certificat auto-signé (l'émetteur est lui-même) avec CN configuré comme cucm-EC.steven.lab. L'extension -EC indique qu'il s'agit du certificat ECDSA configuré sur CUCM.



Sur CUCM, sous Cisco Unified OS Administration, vous pouvez consulter les certificats en place sous Security > Certificate Management, comme indiqué par exemple ici. Il affiche un certificat différent pour tomcat et tomcat-ECDSA où le tomcat est CA signé (et approuvé par l'Expressway-C) tandis que le certificat tomcat-ECDSA est auto-signé et non approuvé par l'Expressway-C ici.



2. L'Adresse De Connexion (FQDN Ou IP) Ne Figure Pas Dans Le Certificat

Outre le magasin de confiance, le serveur de trafic vérifie également l'adresse de connexion vers laquelle le client MRA effectue la requête. Par exemple, quand vous avez configuré sur CUCM sous **System > Server** votre CUCM avec l'adresse IP (10.48.36.215), alors l'Expressway-C

annonce ceci comme tel au client et les requêtes suivantes du client (proxy via l'Expressway-C) sont ciblées vers cette adresse.

Lorsque cette adresse de connexion particulière n'est pas contenue dans le certificat du serveur, la vérification TLS échoue également et une erreur 502 est générée qui entraîne un échec de connexion MRA, par exemple.

```
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
```

Où c3RldmVuLmxhYi9odHRwcy8xMC40OC4zNi4yMTUvODQ0Mw se traduit (base64 - https://www.base64decode.org/) par steven.lab/https/10.48.36.215/8443, ce qui montre qu'il doit établir la connexion vers 10.48.36.215 comme adresse de connexion plutôt que vers cucm.steven.lab. Comme indiqué dans les captures de paquets, le certificat tomcat CUCM ne contient pas l'adresse IP dans le SAN et l'erreur est donc générée.

Comment le valider facilement

Vous pouvez vérifier si vous êtes confronté à ce changement de comportement facilement avec les étapes suivantes :

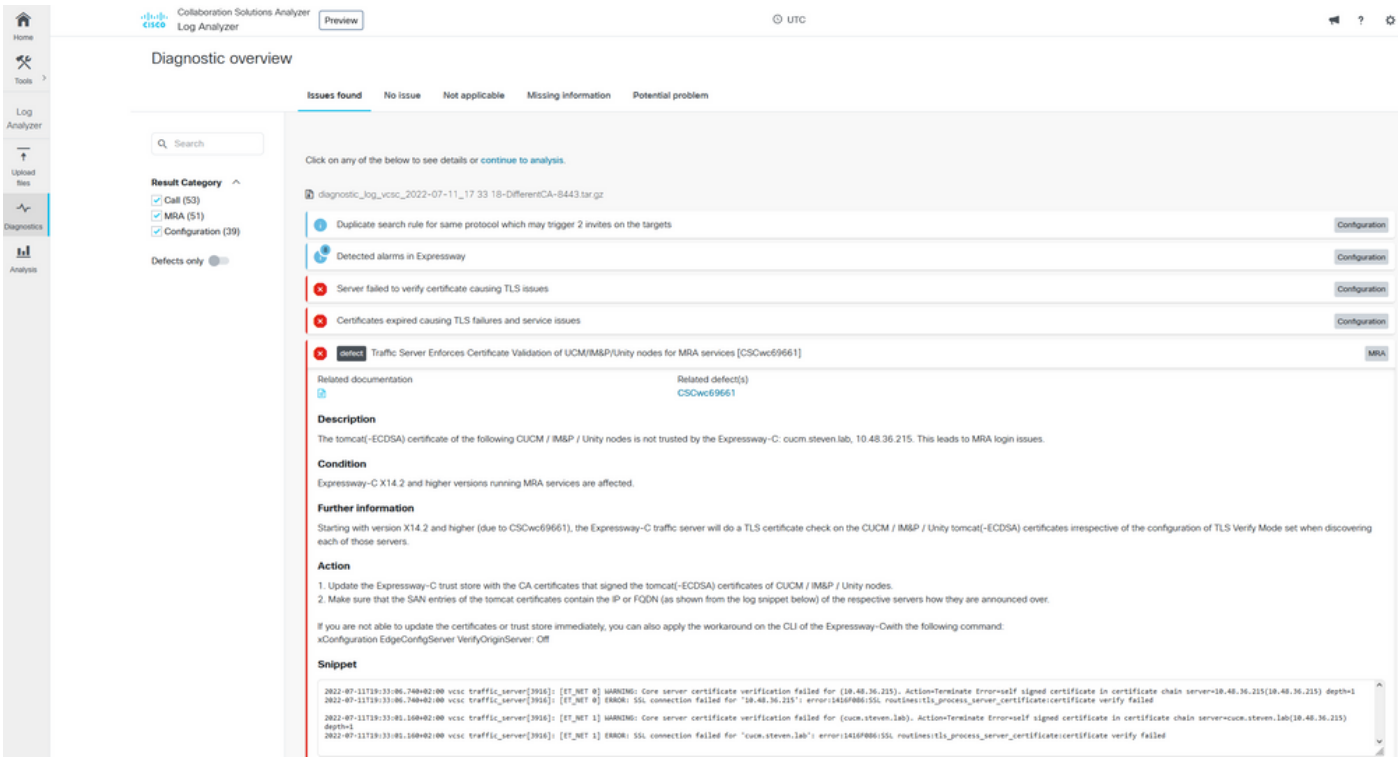
1. Démarrez la journalisation de diagnostic sur le(s) serveur(s) Expressway-E et C (idéalement avec TCPDumps activé) à partir de **Maintenance > Diagnostics > Diagnostic Logging** (dans le cas d'un cluster, il suffit de le démarrer à partir du noeud principal)

2. Tentez une connexion MRA ou testez la fonctionnalité interrompue après la mise à niveau

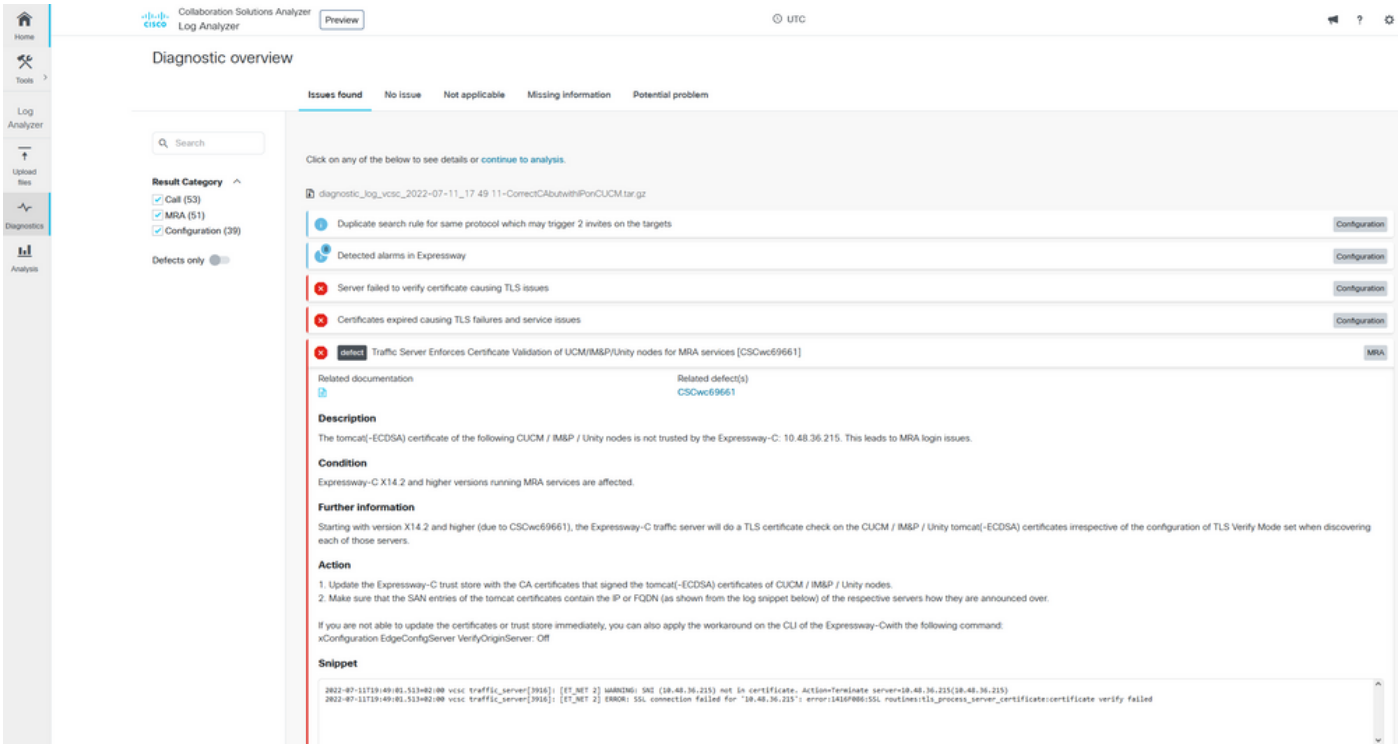
3. Attendez qu'il échoue, puis arrêtez la journalisation de diagnostic sur les serveurs Expressway-E et C (dans le cas d'un cluster, assurez-vous de collecter les journaux de chaque noeud du cluster individuellement)

4. Téléchargez et analysez les journaux sur l'outil Collaboration Solution Analyzer

5. Si vous rencontrez le problème, il récupère les lignes d'avertissement et d'erreur les plus récentes relatives à cette modification pour chacun des serveurs affectés



Signature de diagnostic CA



Signature de diagnostic SNI

Solution

La solution à long terme est de s'assurer que la vérification TLS fonctionne correctement. L'action à effectuer dépend du message d'avertissement affiché.

Lorsque vous observez l'*AVERTISSEMENT : La vérification du certificat du serveur principal a échoué pour (<server-FQDN-or-IP>). Action=Terminate Error=self signed certificate*

server=cucm.steven.lab(10.48.36.215) ***depth=x****, vous devez mettre à jour le magasin de confiance sur les serveurs Expressway-C en conséquence.* Soit avec la chaîne AC qui a signé ce certificat (profondeur > 0) soit avec le certificat auto-signé (profondeur = 0) de **Maintenance > Security > Trusted CA Certificate**. Assurez-vous d'effectuer cette action sur chaque serveur du cluster. Une autre option serait de signer le certificat distant par une autorité de certification connue sur le magasin de confiance d'Expressway-C.

> **Note: Expressway ne permet pas de télécharger deux certificats différents (auto-signés par exemple) dans le magasin de confiance d'Expressway qui ont le même nom commun (CN) selon l'ID de bogue Cisco CSCwa12905. Afin de corriger cela, passez à des certificats signés CA ou mettez à niveau votre CUCM vers la version 14 où vous pouvez réutiliser le même certificat (auto-signé) pour Tomcat et CallManager.**

Lorsque vous observez l'*AVERTISSEMENT : SNI (<server-FQDN-or-IP>) n'est pas dans le* message de *certificat*, alors il indique que ce FQDN ou IP de serveur n'est pas contenu dans le certificat qui a été présenté. Vous pouvez soit adapter le certificat pour inclure ces informations, soit modifier la configuration (comme dans CUCM sur System > Server pour qu'elle corresponde à un élément contenu dans le certificat du serveur), puis actualiser la configuration sur le serveur Expressway-C pour qu'elle soit prise en compte.

La solution à court terme est d'appliquer la solution de contournement comme documenté pour revenir au comportement précédent avant X14.2.0. Vous pouvez effectuer cette opération via l'interface de ligne de commande sur les noeuds du serveur Expressway-C avec la commande nouvellement introduite :

```
xConfiguration EdgeConfigServer VerifyOriginServer: Off
```

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.