Solucionar problemas da verificação de certificado do servidor de tráfego Expressway para serviços MRA introduzidos pelo CSCwc69661

Contents

Introduction

Este documento descreve a alteração de comportamento nas versões do Expressway do X14.2.0 e superior vinculada ao bug da Cisco ID CSCwc6961. Com essa alteração, o servidor de tráfego na plataforma Expressway executa a verificação de certificado do Cisco Unified Communication Manager (CUCM), do Cisco Unified Instant Messaging & Presence (IM&P) e dos nós de servidor do Unity para os serviços de acesso remoto e móvel (MRA). Essa alteração pode levar a falhas de login de MRA após uma atualização na plataforma Expressway.

Prerequisites

Requirements

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Configuração básica do Expressway
- configuração básica MRA

Componentes Utilizados

As informações neste documento são baseadas no Cisco Expressway na versão X14.2 e superior.

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. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Informações de Apoio

O protocolo HTTPS é um protocolo de comunicação seguro que usa o TLS (Transport Layer Security) para criptografar a comunicação. Ele cria esse canal seguro usando um certificado TLS que é trocado no handshake TLS. Dessa forma, ela serve a duas finalidades: autenticação (para saber a quem o participante remoto está conectado) e privacidade (a criptografia). A autenticação protege contra ataques de intermediários e a privacidade impede que os invasores interceptem e interfiram na comunicação.

A verificação de TLS (certificado) é realizada aos olhos da autenticação e permite ter certeza de que você se conectou à parte remota certa. A verificação consiste em dois elementos individuais:

1. Cadeia da Autoridade de Certificação Confiável (AC)

2. Nome Alternativo do Assunto (SAN) ou Nome Comum (CN)

Cadeia de CA confiável

Para que o Expressway-C confie no certificado que o CUCM / IM&P / Unity envia, ele precisa ser capaz de estabelecer um link desse certificado para uma Autoridade de Certificação (CA) de nível superior (raiz) em que ele confia. Esse link, uma hierarquia de certificados que vincula um certificado de entidades a um certificado de CA raiz, é chamado de cadeia de confiança. Para poder verificar essa cadeia de confiança, cada certificado contém dois campos: Emitente (ou 'Emitido por') e Assunto (ou 'Emitido para').

Os certificados de servidor, como o que o CUCM envia para o Expressway-C, têm no campo "Assunto" normalmente seu FQDN (Fully Qualified Domain Name, Nome de domínio totalmente qualificado) no 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
```

Exemplo de um certificado de servidor para CUCM cucm.vngtp.lab. Ele tem o FQDN no atributo CN do campo Assunto junto com outros atributos como País (C), Estado (ST), Local (L), ... Também podemos ver que o certificado do servidor é distribuído (emitido) por uma CA chamada vngtp-ATIVE-DIR-CA.

As CAs de nível superior (CAs raiz) também podem emitir um certificado para se identificarem. Nesse certificado CA raiz, vemos que Emissor e Assunto têm o mesmo valor :

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

Éum certificado passado por uma CA raiz para se identificar.

Em uma situação típica, as CAs raiz não emitem diretamente certificados de servidor. Em vez disso, emitem certificados para outras autoridades de certificação. Essas outras CAs são chamadas de CAs intermediárias. Por sua vez, as autoridades de certificação intermediárias podem emitir diretamente certificados de servidor ou certificados para outras autoridades de certificação intermediárias. Podemos ter uma situação em que um certificado de servidor é emitido pela CA 1 intermediária, que por sua vez recebe um certificado da CA 2 intermediária e assim por diante. Até que a CA intermediária obtenha seu certificado diretamente da CA raiz:

```
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
```

Agora, para que o Expressway-C confie no certificado de servidor que o CUCM envia, ele precisa ser capaz de criar a cadeia de confiança desse certificado de servidor até um certificado de CA raiz. Para que isso aconteça, precisamos carregar o certificado de CA raiz e também todos os certificados de CA intermediários (se houver, o que não é o caso se a CA raiz teria emitido diretamente o certificado de servidor do CUCM) no armazenamento confiável do Expressway-C.

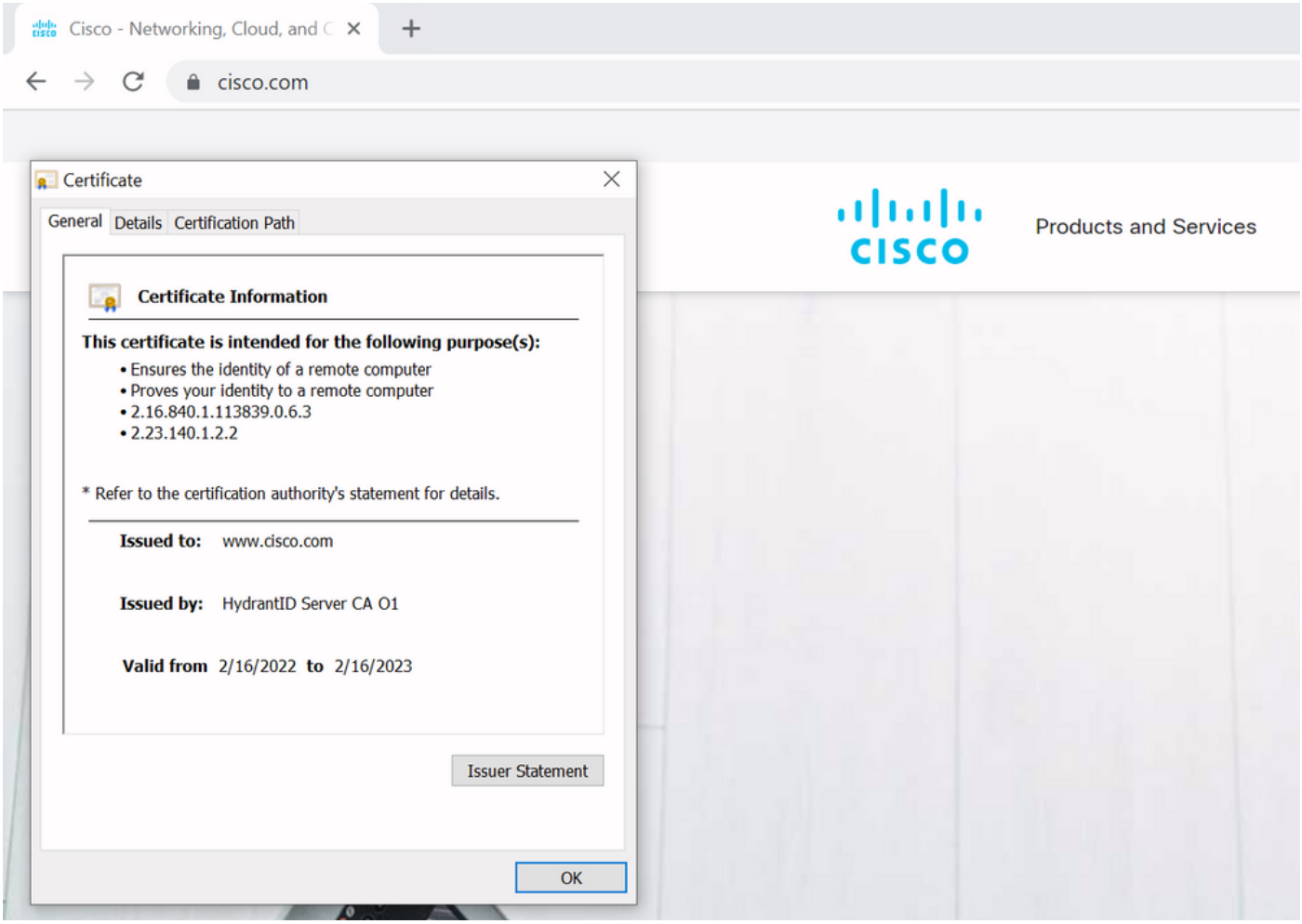
> **Note: Embora os campos Emissor e Assunto sejam fáceis de criar a cadeia de Confiança de forma legível por humanos, o Expressway-C e o CUCM não usam esses campos no certificado. Em vez disso, eles usam os campos 'X509v3 Authority Key Identifier' e 'X509v3 Subject Key Identifier' para criar a cadeia de confiança. Essas chaves contêm identificadores para os certificados que são mais precisos do que para usar os campos Assunto/Emissor : pode haver 2 certificados com os mesmos campos Assunto/Emissor, mas um deles expirou e o outro ainda é válido. Ambos teriam um identificador de chave de assunto X509v3 diferente para que o Expressway/CUCM ainda possa determinar a cadeia de confiança correta.**

Verificação de SAN ou CN

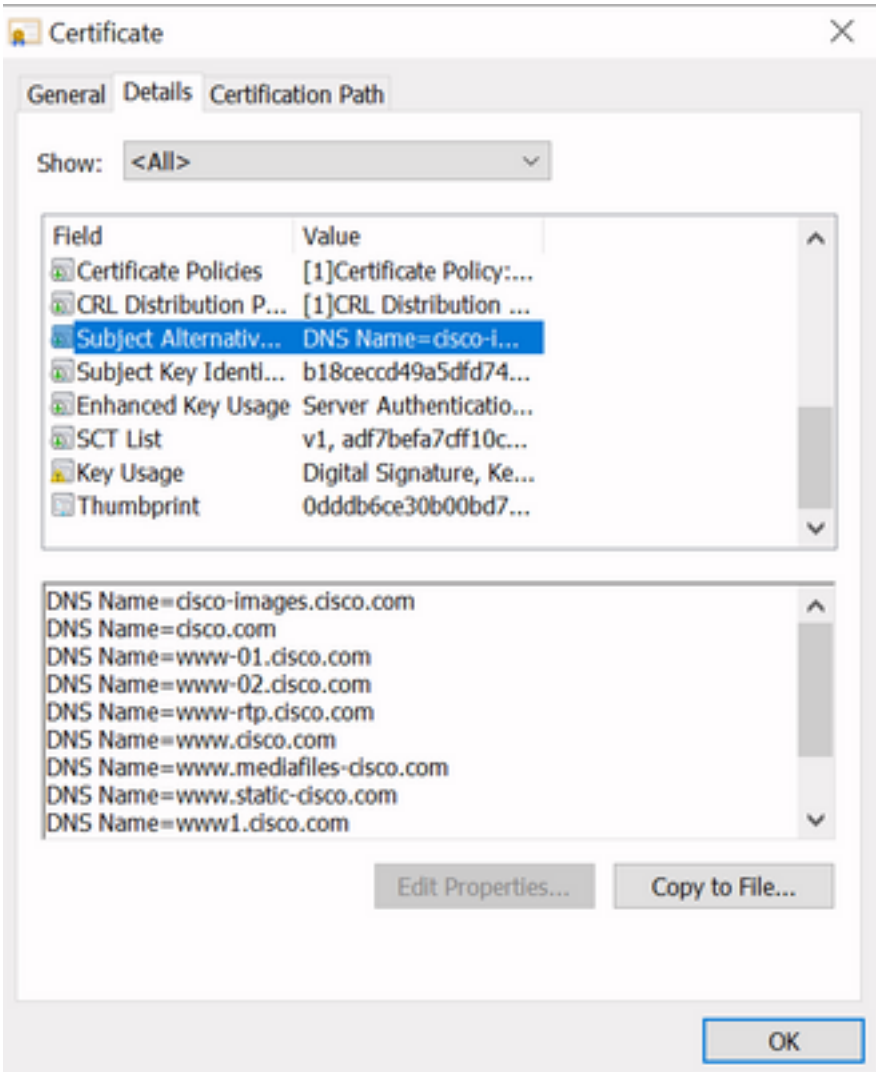
A Etapa 1 faz o check-out do armazenamento confiável, mas qualquer pessoa que tiver um certificado assinado por uma CA no armazenamento confiável será válida. É evidente que isto não é suficiente. Portanto, há uma verificação adicional que valida se o servidor ao qual você se conecta especificamente é realmente o correto. Ele faz isso com base no endereço para o qual o pedido foi feito.

O mesmo tipo de operação acontece em seu navegador, então vamos analisar isso por meio de um exemplo. Se você navegar até https://www.cisco.com, verá um ícone de cadeado ao lado do URL inserido e isso significa que a conexão é confiável. Isso é baseado na cadeia de confiança da CA (da primeira seção) e na verificação SAN ou CN. Se abrirmos o certificado (através do navegador com um clique no ícone de cadeado), você verá que o Nome comum (visto no campo

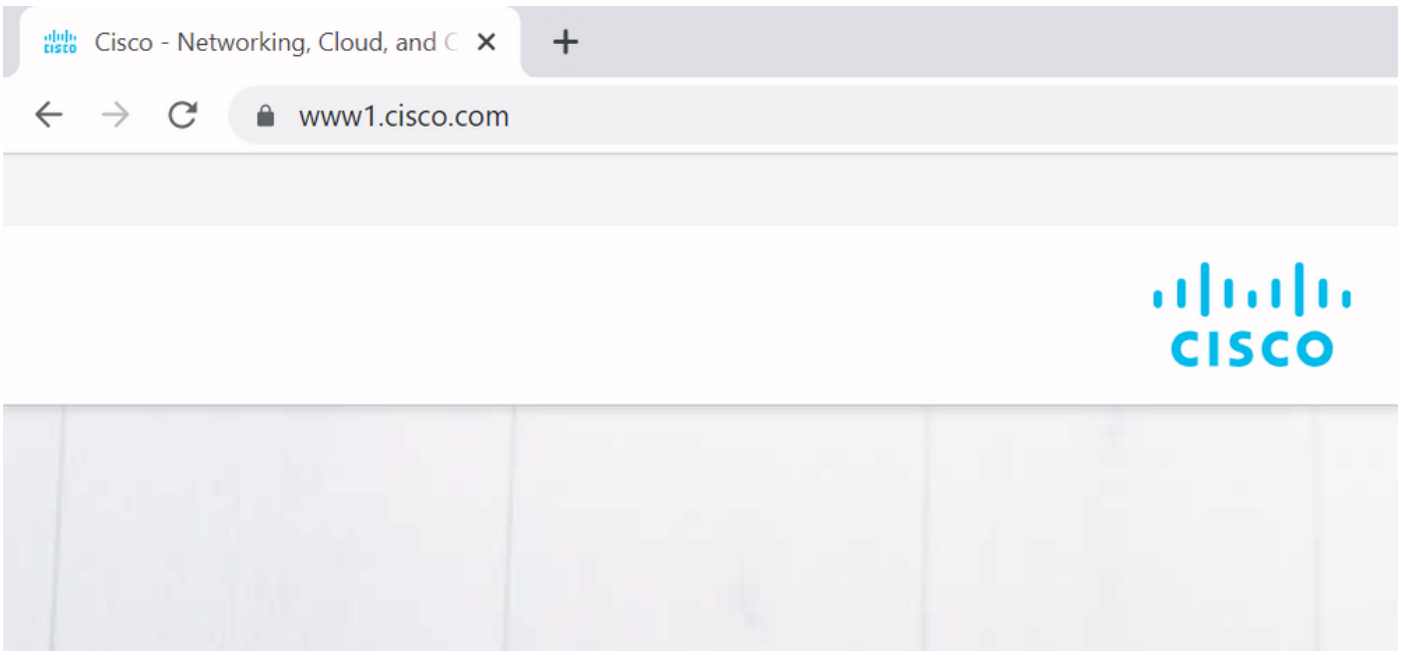
'Emitido para:') está definido como www.cisco.com e que corresponde exatamente ao endereço ao qual queríamos nos conectar. Dessa forma, podemos ter certeza de que nos conectamos ao servidor correto (porque confiamos na CA que assinou o certificado e que executa a verificação antes de entregar o certificado).



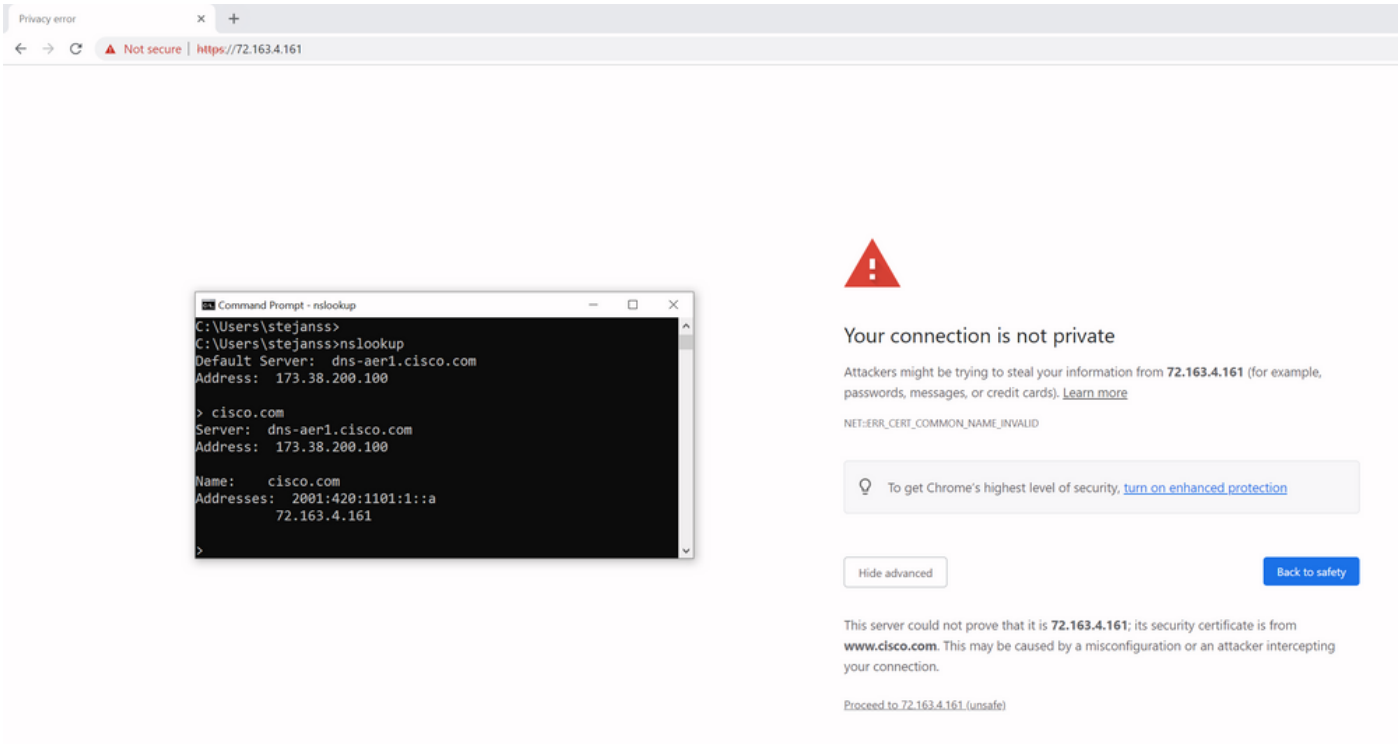
Quando observamos os detalhes do certificado e, em particular, as entradas SAN, vemos que o mesmo se repete, assim como alguns outros FQDNs:



Isso significa que, quando solicitamos a conexão com https://www1.cisco.com, por exemplo, ela também seria exibida como uma conexão segura porque está contida nas entradas SAN.



No entanto, quando não navegamos até https://www.cisco.com, mas diretamente até o endereço IP (https://72.163.4.161), ela não mostra uma conexão segura porque confia na CA que a assinou, mas o certificado apresentado a nós não contém o endereço (72.163.4.161) que usamos para conexão com o servidor.



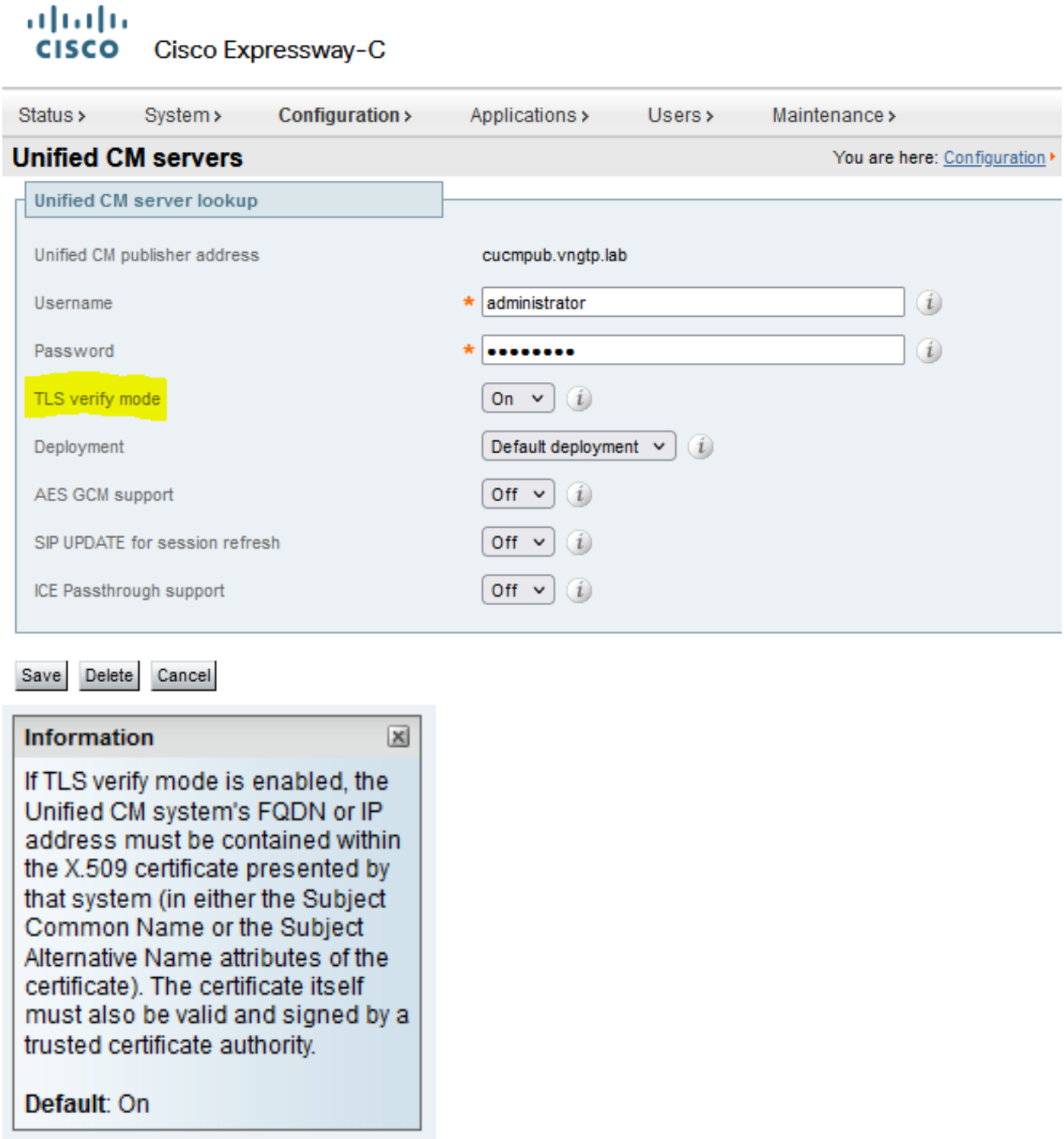
No navegador, você pode ignorar isso, mas essa é uma configuração que você pode habilitar em conexões TLS que um desvio não é permitido. Portanto, é importante que seus certificados contenham os nomes CN ou SAN corretos que a parte remota planeja usar para se conectar a eles.

Alteração de comportamento

Os serviços MRA dependem muito de várias conexões HTTPS nos Expressways em direção aos servidores CUCM / IM&P / Unity para se autenticar corretamente e coletar as informações certas específicas para o cliente que faz login. Essa comunicação geralmente acontece nas portas 8443 e 6972.

Versões anteriores a X14.2.0

Em versões anteriores a X14.2.0, o servidor de tráfego no Expressway-C que lida com essas conexões HTTPS seguras não verificou o certificado apresentado pela extremidade remota. Isso pode levar a ataques de intermediários. Na configuração MRA, há uma opção para verificação de certificado TLS pela configuração do 'Modo de verificação TLS' como 'Ativado' quando você adicionaria servidores CUCM / IM&P / Unity em **Configuração > Comunicações Unificadas > servidores Unified CM / nós de Serviço IM e Presence / servidores Unity Connection**. A opção de configuração e a caixa de informações relevantes são mostradas como exemplo, indicando que ela verifica o FQDN ou o IP na SAN, bem como a validade do certificado e se ele está assinado por uma CA confiável.



Esta verificação de certificado TLS só é feita na descoberta dos servidores CUCM / IM&P / Unity e não no momento em que, durante o login MRA, os vários servidores são consultados. Uma primeira desvantagem dessa configuração é que ela apenas verifica o endereço do publicador adicionado. Ele não valida se o certificado nos nós do assinante foi configurado corretamente, pois recupera as informações do nó do assinante (FQDN ou IP) do banco de dados do nó do publicador. Uma segunda desvantagem dessa configuração também é que o que é anunciado para os clientes MRA como as informações de conexão podem ser diferentes do endereço do publicador que foi colocado na configuração Expressway-C. Por exemplo, no CUCM, em **System > Server**, você pode anunciar o servidor com um endereço IP (10.48.36.215, por exemplo) e isso é usado pelos clientes MRA (através da conexão do Expressway com proxy); no entanto, você pode adicionar o CUCM no Expressway-C com o FQDN de cucm.steven.lab. Suponha que o certificado tomcat do CUCM contenha cucm.steven.lab como entrada de SAN, mas não o endereço IP, então a descoberta com 'TLS Verify Mode' definido como 'On' é bem-sucedida, mas

as comunicações reais dos clientes MRA podem ter como destino um FQDN ou IP diferente e, portanto, falhar na verificação de TLS.

Versões do X14.2.0 e posterior

A partir da versão X14.2.0, o servidor Expressway executa a verificação de certificado TLS para cada solicitação HTTPS feita através do servidor de tráfego. Isso significa que ele também executa isso quando o 'Modo de verificação TLS' está definido como 'Desligado' durante a descoberta dos nós CUCM / IM&P / Unity. Quando a verificação não é bem-sucedida, o handshake TLS não é concluído e a solicitação falha, o que pode levar à perda de funcionalidade, como problemas de redundância ou failover ou falhas de login completas, por exemplo. Além disso, com o 'Modo de verificação de TLS' definido como 'Ativado', ele não garante que todas as conexões funcionem bem, conforme abordado no exemplo mais adiante.

Além do padrão na verificação TLS, há também uma alteração introduzida no X14.2 que anuncia uma ordem de preferência diferente para a lista de cifras. Isso pode causar conexões TLS inesperadas após uma atualização de software, pois pode acontecer que, antes da atualização, ele solicite o certificado Cisco Tomcat ou Cisco CallManager do CUCM (ou qualquer outro produto que tenha um certificado separado para o algoritmo ECDSA), mas que, após a atualização, ele solicite a variante ECDSA. Os certificados Cisco Tomcat-ECDSA ou Cisco CallManager-ECDSA podem ser assinados por uma CA diferente ou apenas certificados ainda autoassinados (o padrão).

Há duas maneiras de a verificação de TLS falhar neste cenário, que serão abordadas em detalhes posteriormente:

1. A autoridade de certificação que assinou o certificado remoto não é confiável

a. Certificado autoassinado

b. Certificado assinado por CA desconhecida

2. O Endereço de Conexão (FQDN ou IP) não está contido no certificado

Solucionar problemas de cenários

Os próximos cenários mostram um cenário semelhante em um ambiente de laboratório onde o login de MRA falhou após uma atualização do Expressway de X14.0.7 para X14.2. Eles compartilham semelhanças nos logs, no entanto, a resolução é diferente. Os logs são coletados pelo log de diagnóstico (de **Manutenção > Diagnóstico > Log de diagnóstico**) que foi iniciado antes do logon de MRA e interrompido depois que o logon de MRA falhou. Nenhum log de depuração adicional foi habilitado para ele.

1. A autoridade de certificação que assinou o certificado remoto não é confiável

O certificado remoto pode ser assinado por uma CA que não esteja incluída no armazenamento confiável do Expressway-C ou pode ser um certificado autoassinado (em essência, uma CA também) que não é adicionado no armazenamento confiável do servidor Expressway-C.

No exemplo aqui, você pode observar que as solicitações que vão para o CUCM (10.48.36.215 - cucm.steven.lab) são tratadas corretamente na porta 8443 (resposta 200 OK), mas ela gera um

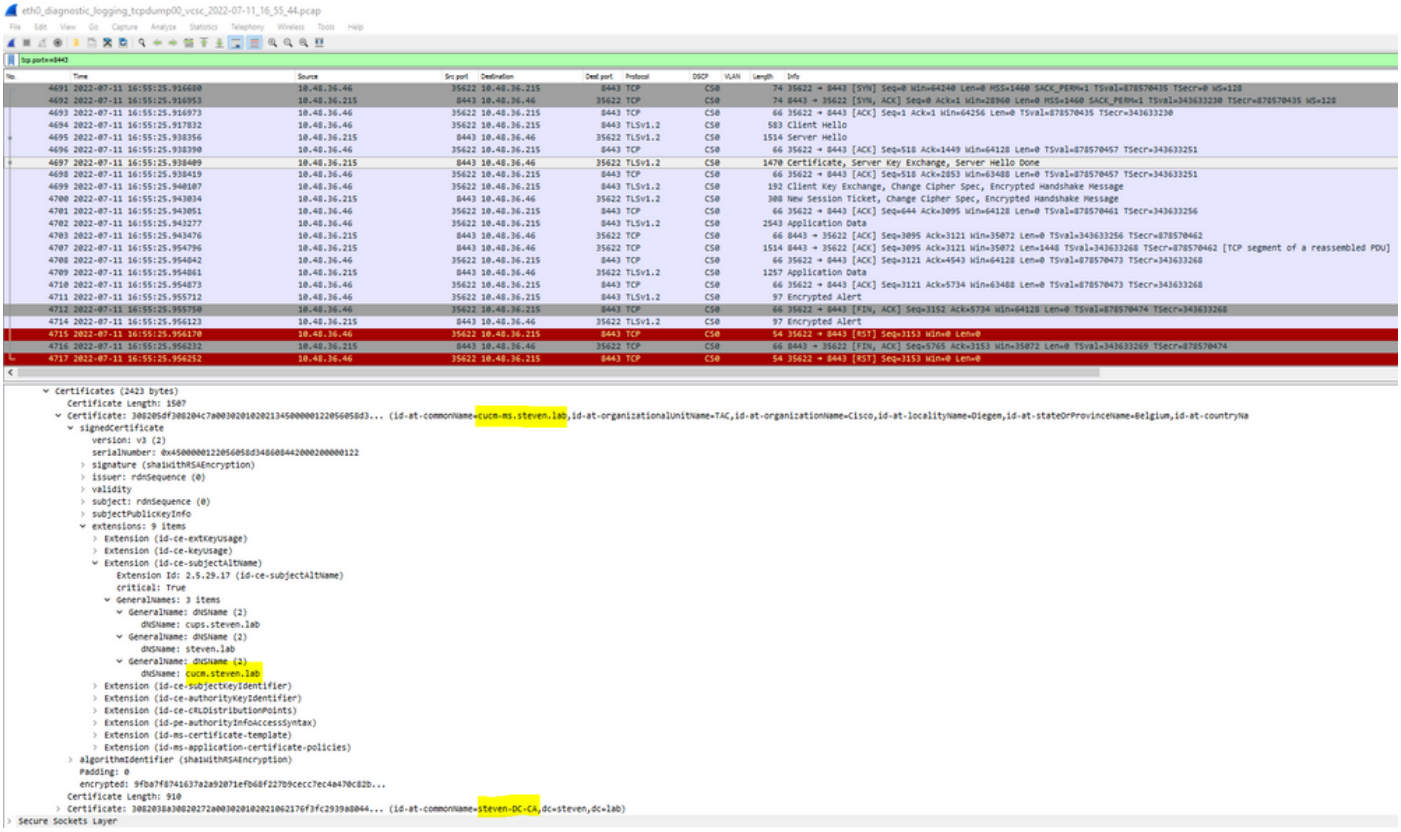
erro (resposta 502) na porta 6972 para a conexão TFTP.

```
===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"
```

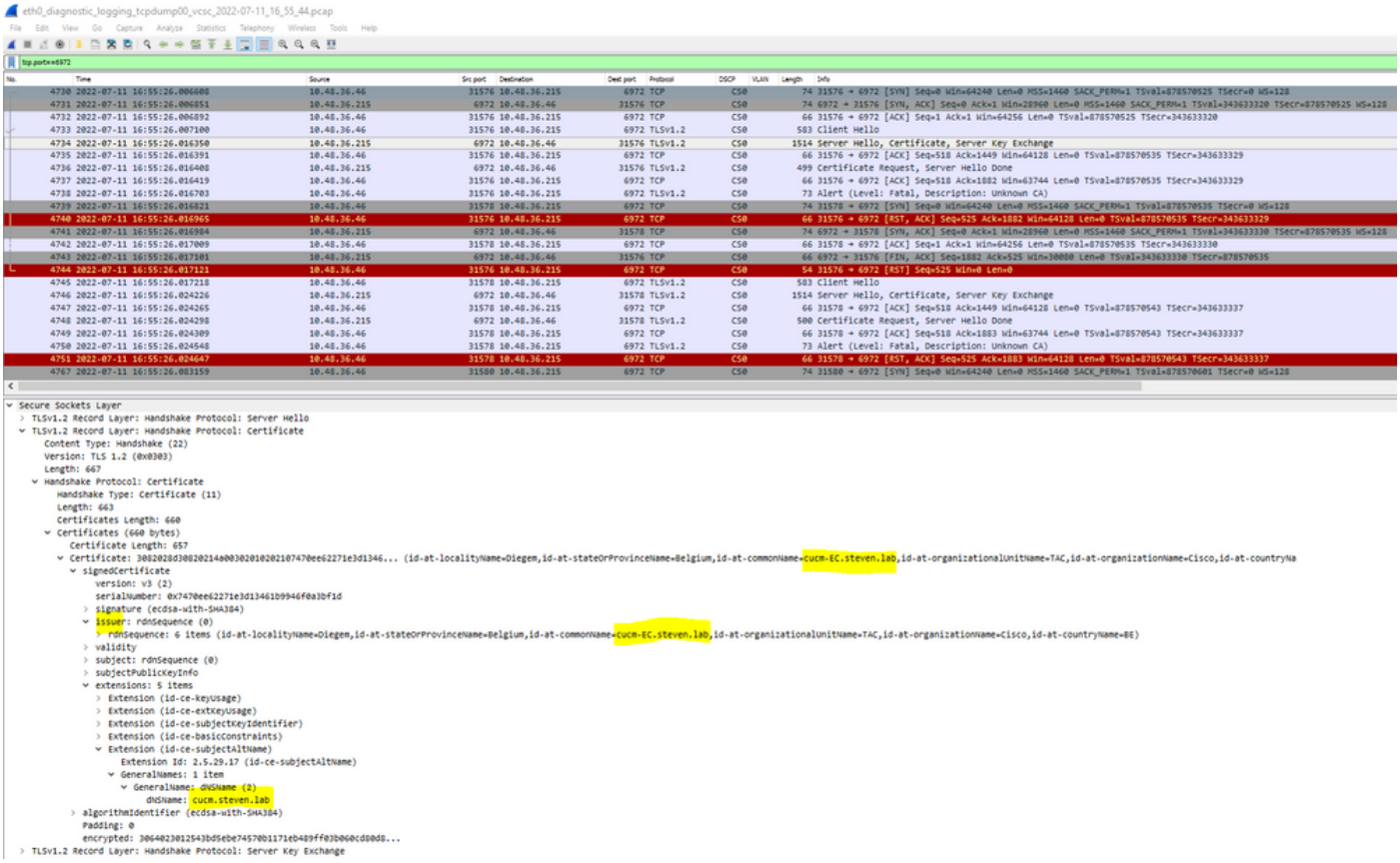
O erro de 'falha na verificação do certificado' indica o fato de que o Expressway-C não pôde validar o handshake TLS. A razão para isso é mostrada na linha de aviso, pois indica um certificado autoassinado. Se a profundidade for mostrada como 0, é um certificado autoassinado. Quando a profundidade é maior que 0, isso significa que ela tem uma cadeia de certificados e, portanto, é assinada por uma CA desconhecida (da perspectiva do Expressway-C).

Quando observamos o arquivo pcap que foi coletado nos carimbos de data e hora mencionados

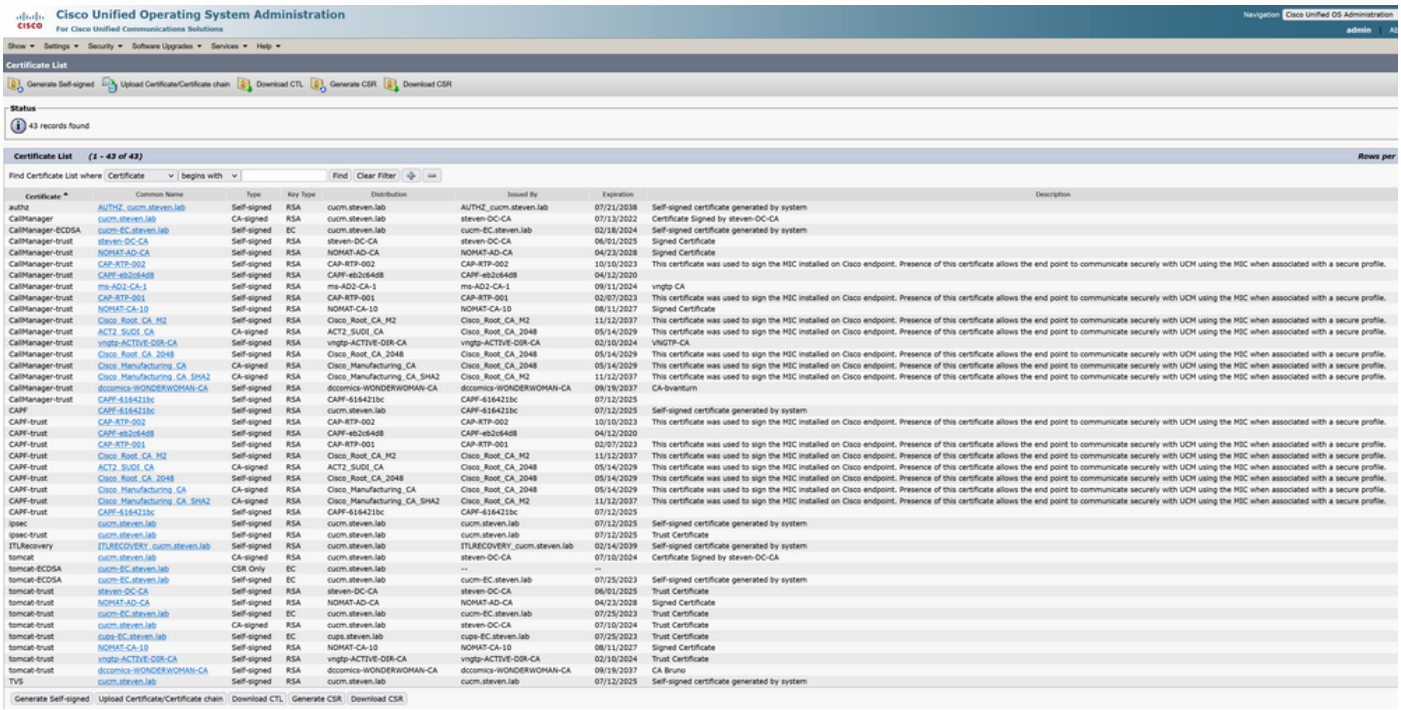
nos logs de texto, você pode ver que o CUCM apresenta o certificado com CN como cucm-ms.steven.lab (e cucm.steven.lab como SAN) assinado por steven-DC-CA ao Expressway-C na porta 8443.



Mas quando inspecionamos o certificado apresentado na porta 6972, você pode ver que ele é um certificado autoassinado (o próprio Emissor) com CN configurado como cucm-EC.steven.lab. A extensão -EC dá a indicação de que este é o certificado ECDSA configurado no CUCM.



No CUCM, em Cisco Unified OS Administration, você pode ver os certificados em vigor em Segurança > Gerenciamento de certificado, como mostrado aqui, por exemplo. Ele mostra um certificado diferente para tomcat e tomcat-ECDSA em que o tomcat é assinado pela CA (e confiável pelo Expressway-C) enquanto o certificado tomcat-ECDSA é autoassinado e não confiável pelo Expressway-C aqui.



2. O endereço de conexão (FQDN ou IP) não está contido no certificado

Além do armazenamento confiável, o servidor de tráfego também verifica o endereço de conexão para o qual o cliente MRA faz a solicitação. Por exemplo, quando você tiver configurado no CUCM em **System > Server** seu CUCM com o endereço IP (10.48.36.215), o Expressway-C anunciará isso como tal ao cliente e as solicitações subsequentes do cliente (com proxy através do Expressway-C) serão direcionadas a esse endereço.

Quando esse endereço de conexão específico não está contido no certificado do servidor, a verificação TLS também falha e um erro 502 é lançado, resultando em falha de login de MRA, por exemplo.

```
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
```

Em que c3RldmVuLmxhYi9odHRwcy8xMC40OC4zNi4yMTUvODQ0Mw converte (base64 - https://www.base64decode.org/) para steven.lab/https/10.48.36.215/8443, que mostra que deve fazer a conexão em direção a 10.48.36.215 como o endereço de conexão, em vez de para cucm.steven.lab. Como mostrado nas capturas de pacote, o certificado tomcat CUCM não contém o endereço IP na SAN e, portanto, o erro é lançado.

Como validá-lo facilmente

Épossível validar se você se depara com essa mudança de comportamento facilmente com as próximas etapas:

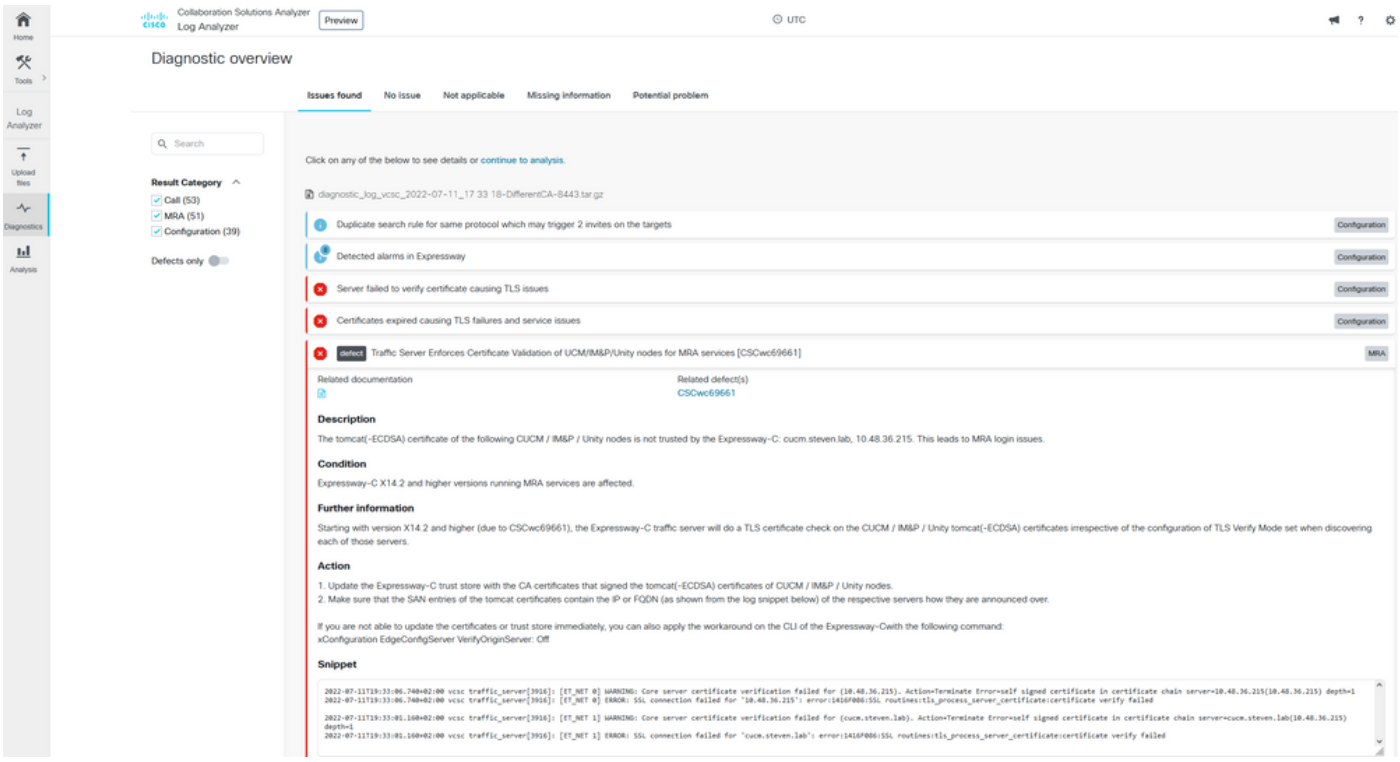
1. Inicie o log de diagnóstico no(s) servidor(es) Expressway-E e C (idealmente com TCPDumps ativados) de **Manutenção > Diagnóstico > Log de Diagnóstico** (no caso de um cluster, é suficiente iniciá-lo a partir do nó mestre)

2. Tente um login de MRA ou teste a funcionalidade interrompida após a atualização

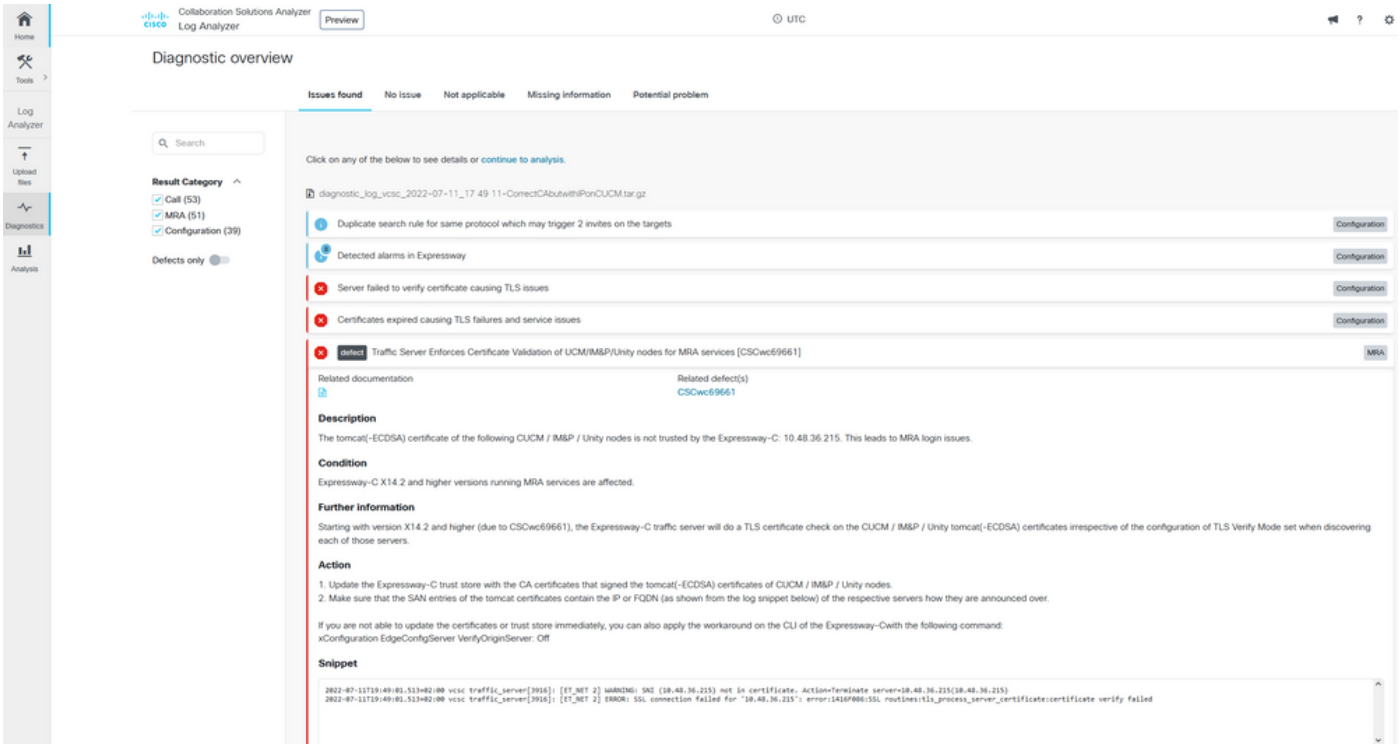
3. Aguarde até que haja falha e pare o log de diagnóstico no(s) servidor(es) Expressway-E e C (no caso de um cluster, certifique-se de coletar os logs de cada nó individual do cluster individualmente)

4. Carregue e analise os logs na ferramenta Collaboration Solution Analyzer

5. Se você encontrar o problema, ele selecionará as linhas de aviso e de erro mais recentes relacionadas a essa alteração para cada um dos servidores afetados



assinatura de diagnóstico de CA



Assinatura de diagnóstico SNI

Solução

A solução de longo prazo é garantir que a verificação TLS funcione bem. A ação a ser executada depende da mensagem de aviso exibida.

Quando você observar o *AVISO: Falha na verificação do certificado do servidor núcleo para (<server-FQDN-or-IP>).* Action=Terminate Error=self signed certificate server=cucm.steven.lab(10.48.36.215) ***depth=x, então você precisa atualizar o armazenamento confiável nos servidores Expressway-C adequadamente.*** Com a cadeia de CAs que assinou este certificado (profundidade > 0) ou com o certificado autoassinado (profundidade = 0) em **Manutenção > Segurança > Certificado de CA Confiável**. Certifique-se de executar esta ação em todos os servidores do cluster. Outra opção seria assinar o certificado remoto por uma CA conhecida no repositório de confiança do Expressway-C.

Quando você observar o *AVISO: SNI (<server-FQDN-or-IP>) não na* mensagem *de certificado*, indica que esse FQDN ou IP do servidor não está contido no certificado que foi apresentado. Você pode adaptar o certificado para incluir essas informações ou pode modificar a configuração (como no CUCM em Sistema > Servidor para algo que esteja contido no certificado do servidor) e, em seguida, atualizar a configuração no servidor Expressway-C para que ela seja levada em conta.

A solução de curto prazo é aplicar a solução alternativa conforme documentado para fazer fallback para o comportamento anterior antes de X14.2.0. Você pode executar isso através da CLI nos nós do servidor Expressway-C com o comando recém-introduzido:

```
xConfiguration EdgeConfigServer VerifyOriginServer: Off
```

Sobre esta tradução
A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês (link fornecido) seja sempre consultado.