排除MRA服务的Expressway流量服务器证书验证故障

目录

简介

本文档介绍与思科漏洞ID CSCwc69661或思科漏洞ID CSCwa25108关联的Expressway X14.2.0及更高版本上的行为更改。

先决条件

要求

Cisco 建议您了解以下主题：

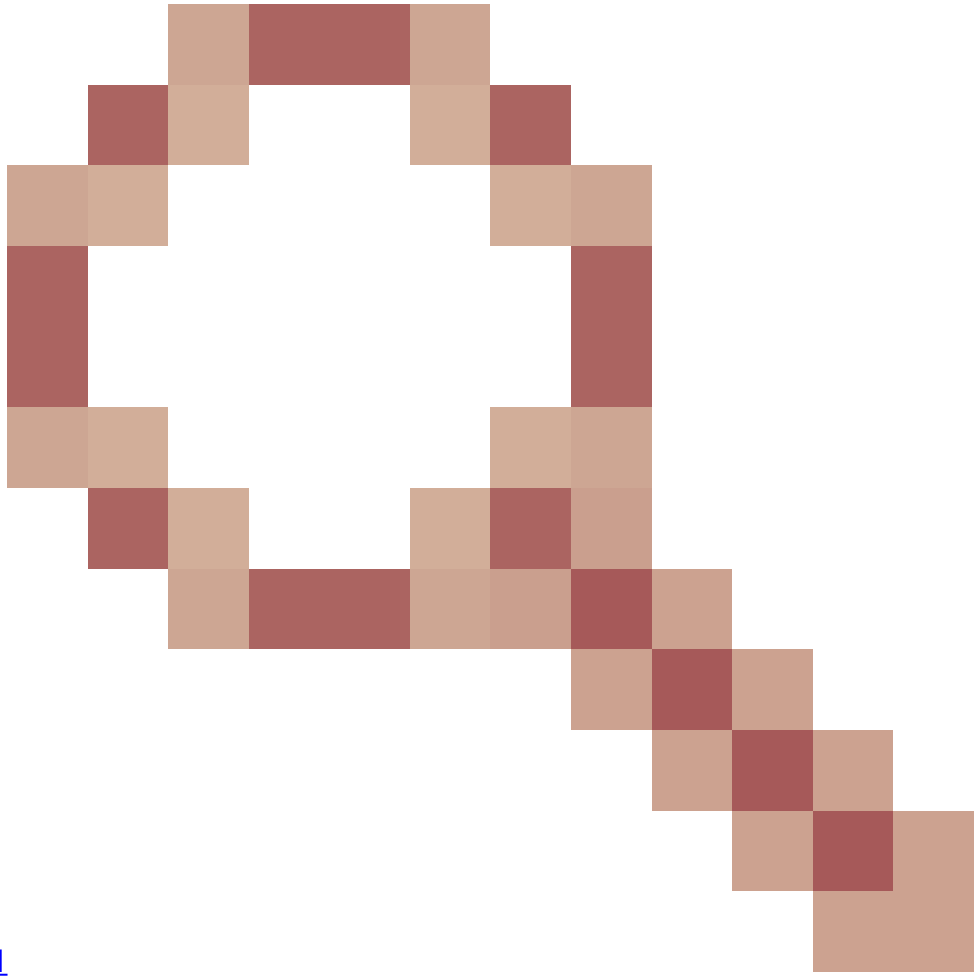
- Expressway基本配置
- MRA基本配置

使用的组件

本文档中的信息基于X14.2及更高版本上的Cisco Expressway。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息



通过思科漏洞ID CSCwc69661
或思科漏洞ID CSCwa25108标记的行为更改，Expressway平台上的流量服务器执行针对移动和远程访问(MRA)服务的Cisco Unified Communication Manager (CUCM)、思科统一即时消息和在线状态(IM&P)以及Unity服务器节点的证书验证。此更改可能会导致Expressway平台升级后的MRA登录失败。

安全超文本传输协议(HTTPS)是一种使用传输层安全(TLS)加密通信的安全通信协议。它使用TLS握手过程中交换的TLS证书创建此安全通道。此服务器有两个用途：身份验证（了解您连接的远程方）和隐私（加密）。身份验证可防止中间人攻击，并且隐私保护可防止攻击者窃听和篡改通信。

TLS（证书）验证在看到身份验证时执行，并允许您确保您已连接到正确的远程方。验证包括两项：

1. 受信任的证书颁发机构(CA)链

2. 主题备用名称(SAN)或公用名称(CN)

可信CA链

要使Expressway-C信任CUCM/IM&P/Unity发送的证书，它需要能够建立从该证书到其信任的顶级（根）证书颁发机构(CA)的链接。此类链接是将实体证书链接到根CA证书的证书层次结构，称为信任链。为了能够验证此类信任链，每个证书包含两个字段：Issuer（或“Issued by”）和Subject（或“Issued To”）。

服务器证书（例如CUCM发送到Expressway-C的证书）在“Subject”字段中通常具有其在CN中的完全限定域名(FQDN)：

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

CUCM cucm.vngtp.lab的服务器证书示例。它在Subject字段的CN属性中具有FQDN，同时还具有其他属性，例如Country (C)、State (ST)、Location (L)……我们还可以看到服务器证书由名为vngtp-ACTIVE-DIR-CA的CA分发（颁发）。

顶级CA（根CA）也可以颁发证书来标识自己。在这样的根CA证书中，我们可以看到Issuer和Subject具有相同的值：

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

它是根CA分发的证书，用于标识自身。

在典型情况下，根CA不会直接颁发服务器证书。相反，它们会为其他CA颁发证书。此类其他CA然后称为中间CA。反过来，中间CA可以直接为其他中间CA颁发服务器证书或证书。我们可能会遇到服务器证书由中间CA 1颁发，而中间CA 1又从中间CA 2等获取证书的情况。直到最后中间CA直接从根CA获取其证书：

```
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
```

现在，为了使Expressway-C信任CUCM发送的服务器证书，它需要能够构建从该服务器证书一直到根CA证书的信任链。为此，我们需要在Expressway-C的信任存储中上传根CA证书和所有中间CA证书（如果有，如果根CA会直接颁发CUCM的服务器证书则不会出现这种情况）。

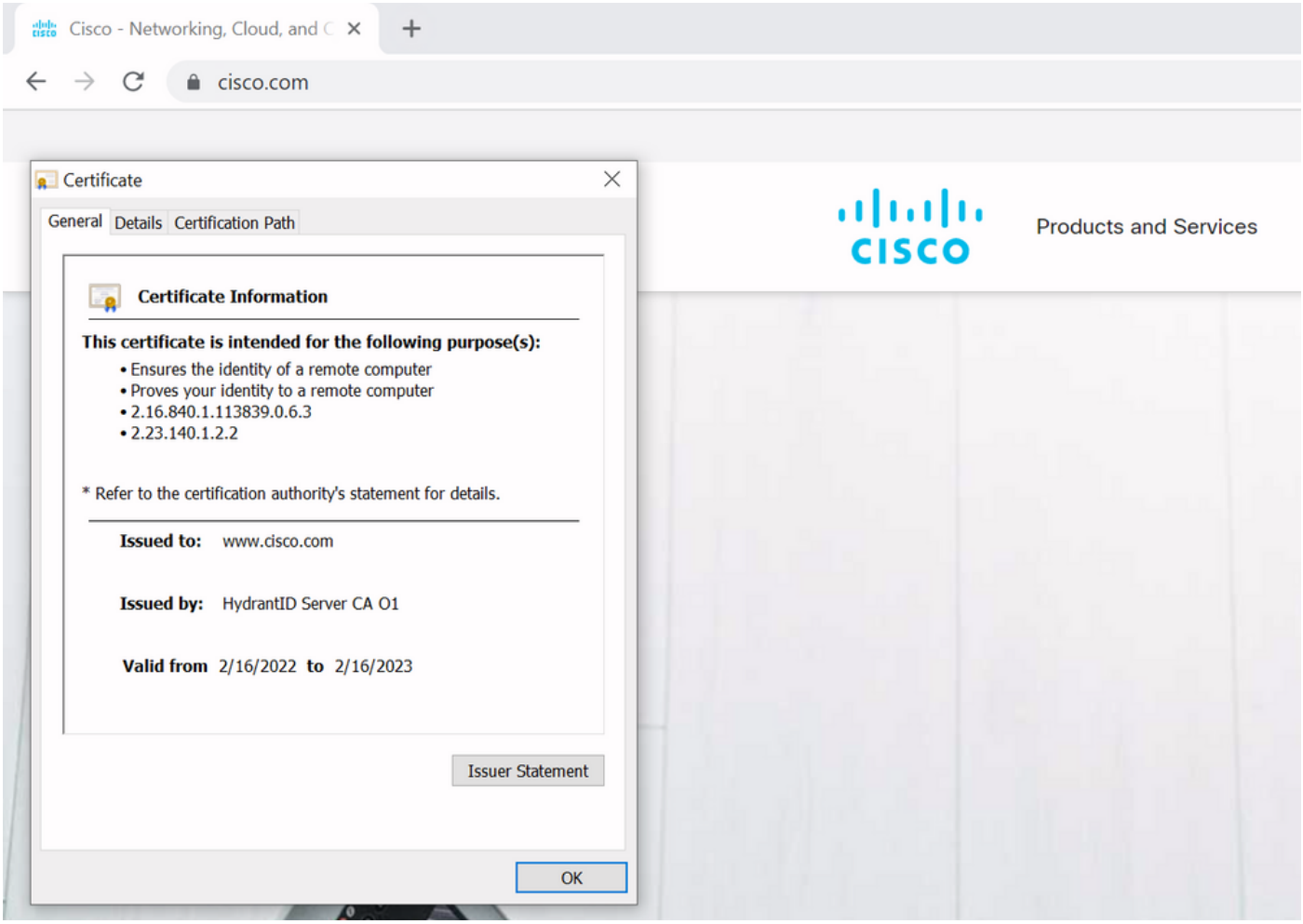
注意：虽然Issuer和Subject字段易于以易于阅读的方式构建信任链，但CUCM在证书中并不使用这些字段。相反，它使用“X509v3 Authority Key Identifier”和“X509v3 Subject Key Identifier”字段构建信任链。这些密钥包含比使用Subject/Issuer字段更准确的证书标识符：可以有两个具有相同Subject/Issuer字段的证书，但其中一个已过期，另一个仍然有效。它们都有不同的X509v3主题密钥标识符，因此CUCM仍可确定正确的信任链。

但根据思科漏洞ID CSCwa12905，Expressway不会出现这种情况，也不能将两个不同（例如自签名）的证书上传到具有相同公用名(CN)的Expressway信任库中。对此进行更正的方法是CA签名证书或对其使用不同的通用名称，或者查看它始终使用相同的证书（可能通过CUCM 14中的重复使用证书功能）。

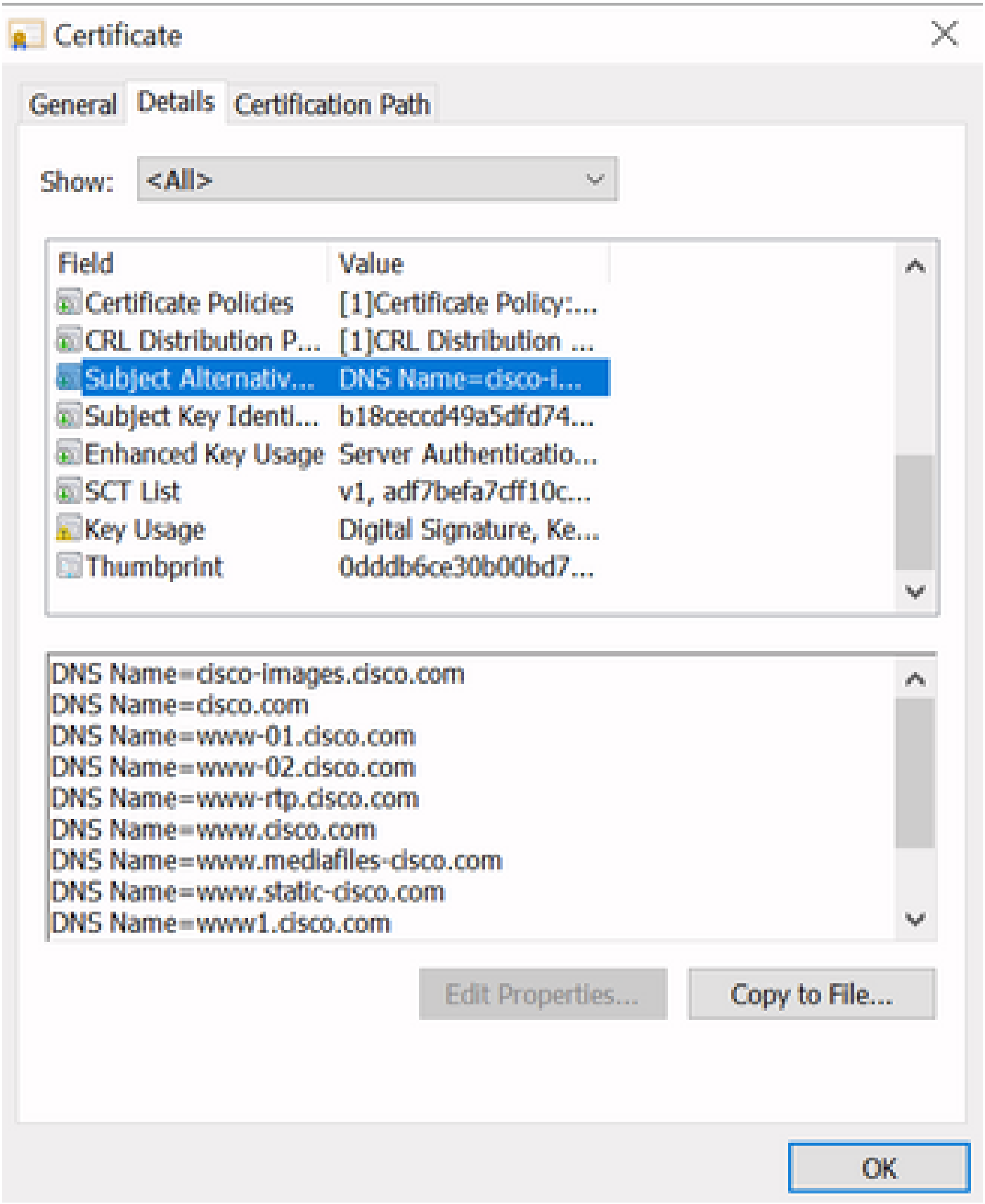
SAN或CN检查

第1步检查信任库，但是，拥有由信任库中的CA签名的证书的任何人到那时都是有效的。这显然不够。因此，还会进行额外的检查，以验证您专门连接的服务器是否正确。它根据发出请求的地址执行此操作。

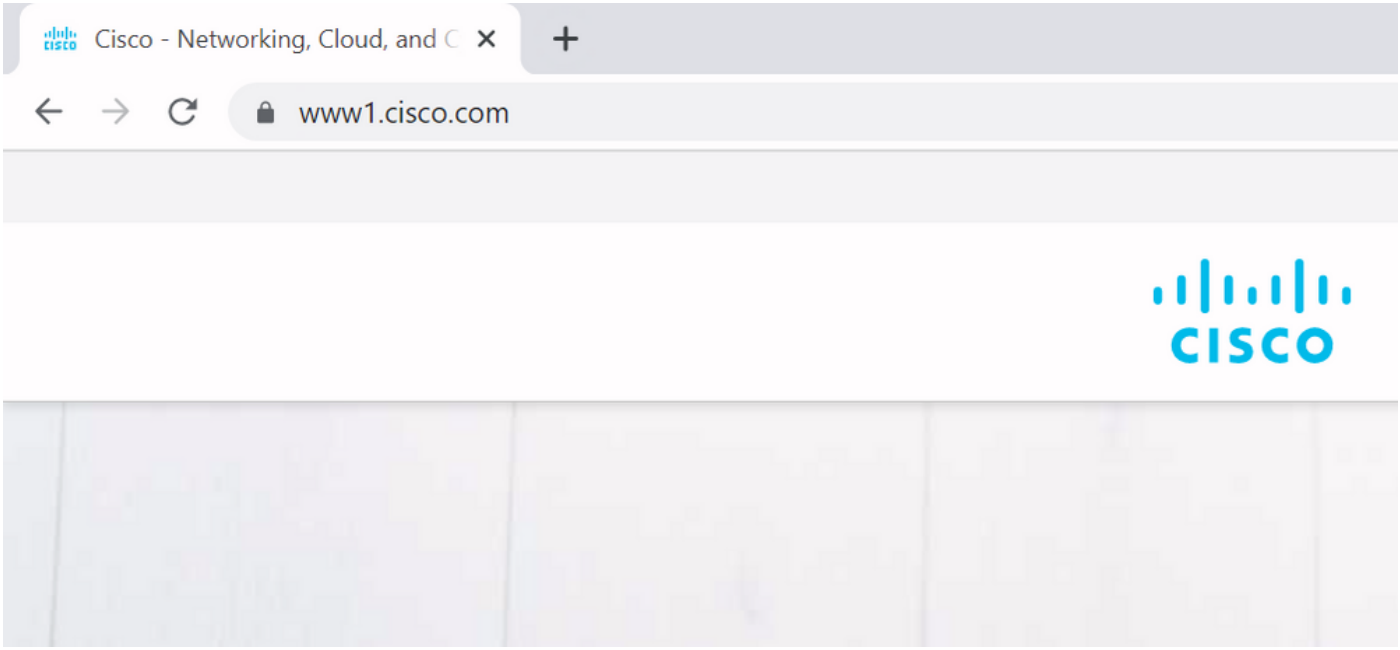
在浏览器中也会发生同样的操作，因此让我们通过一个示例来了解这一点。如果您浏览 https://www.cisco.com，在您输入的URL旁边看到一个锁图标，表示它是受信任的连接。这既基于CA信任链（来自第一部分），也基于SAN或CN检查。如果我们打开证书（通过浏览器单击锁图标），您会看到“常用名”（在“Issued to：”字段中看到）设置为www.cisco.com，并且与我们想要连接的地址完全一致。这样，可以确保我们连接到正确的服务器（因为我们信任签署证书并在分发证书之前执行验证的CA）。



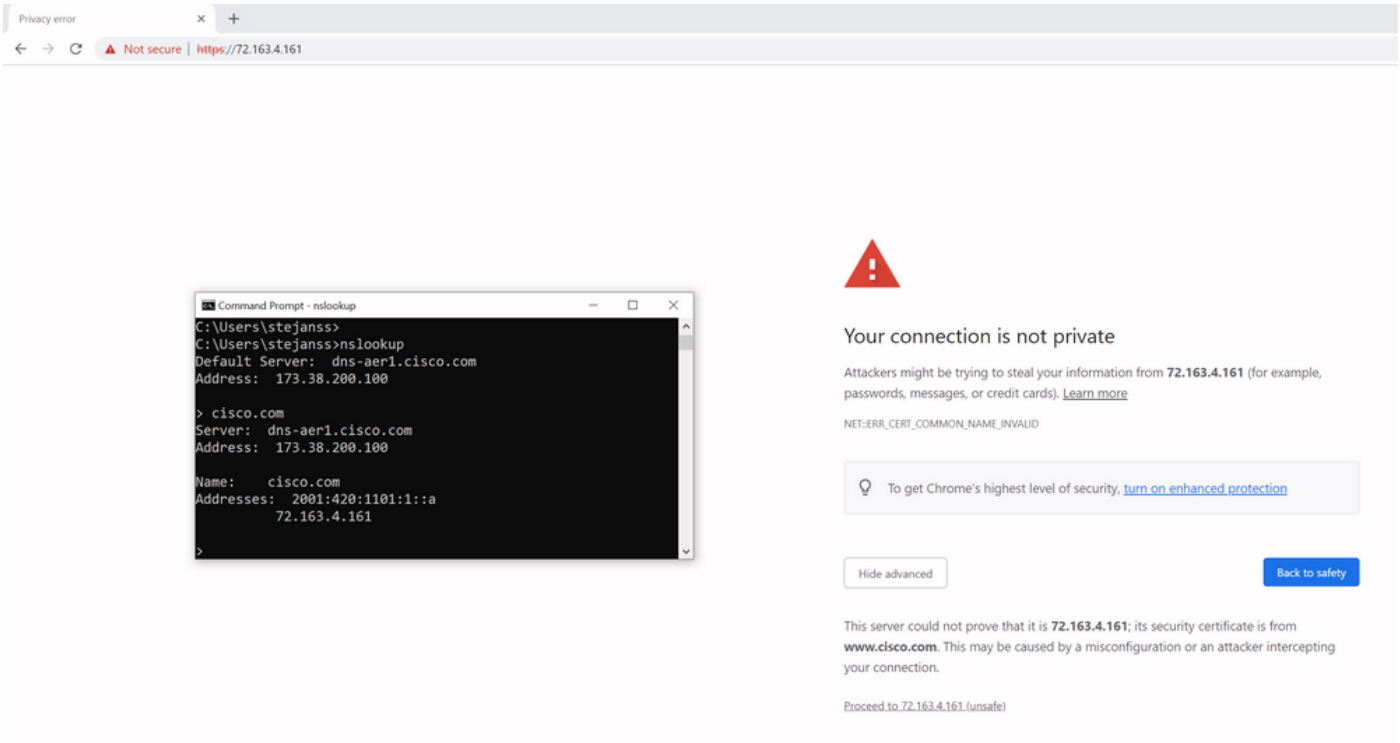
当我们查看证书的详细信息（尤其是SAN条目）时，我们会看到重复了相同的内容以及某些其他FQDN：



例如，这意味着当我们请求连接到https://www1.cisco.com时，由于它包含在SAN条目中，因此也会显示为安全连接。



但是，当我们不浏览https://www.cisco.com而直接浏览IP地址(https://72.163.4.161)时，它不会显示安全连接，因为它信任对其进行签名的CA，但提供给我们的证书不包含用于连接到服务器的地址(72.163.4.161)。



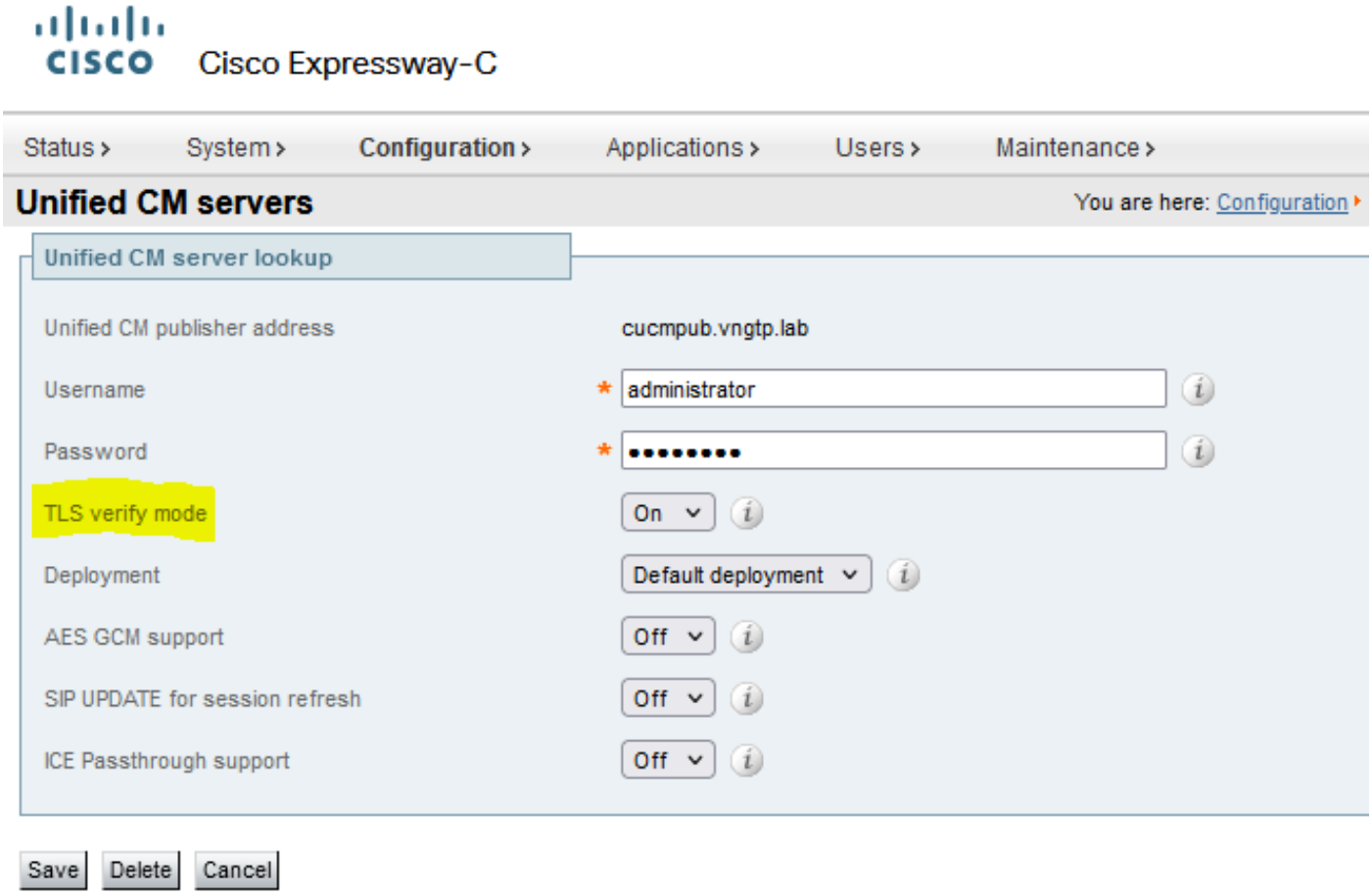
在浏览器中，您可以绕过此设置，但是可以在TLS连接上启用不允许绕过此设置的设置。因此，您的证书必须包含远程方计划用来连接它的CN或SAN名称，这一点非常重要。

行为更改

MRA服务严重依赖于Expressway上指向CUCM/IM&P/Unity服务器的几个HTTPS连接，以便正确进行身份验证并收集特定于登录客户端的正确信息。此通信通常通过端口8443和6972进行。

低于X14.2.0的版本

在低于X14.2.0的版本中，Expressway-C上处理这些安全HTTPS连接的流量服务器不验证远程端提供的证书。这可能导致中间人攻击。在MRA配置上，当您需要在Configuration > Unified Communications > Unified CM servers / IM and Presence Service nodes / Unity Connection servers下添加CUCM / IM&P / Unity服务器时，有一个通过将“TLS验证模式”配置为“开”来验证TLS证书的选项。配置选项和相关信息框以示例形式显示，表明它确实验证了SAN中的FQDN或IP以及证书的有效性以及证书是否由受信任CA签署。



此TLS证书验证检查仅在发现CUCM/IM&P/Unity服务器时完成，而不是在MRA登录期间查询各种服务器时完成。此配置的第一个缺点是，它只验证您添加的发布者地址。它不会验证订用服务器节点上的证书是否已正确设置，因为它从发布服务器节点的数据库中检索订用服务器节点信息（FQDN或IP）。此配置的第二个缺点是，由于连接信息可能不同于Expressway-C配置中设置的发布方地址，因此通告给MRA客户端的内容。例如，在CUCM上，在System > Server下，您可以使用IP地址（例如10.48.36.215）向外部通告服务器，然后由MRA客户端（通过代理的Expressway连接）使用，但是您可以在Expressway-C上使用FQDN cucm.steven.lab添加CUCM。因此，假设CUCM的tomcat证书包含cucm.steven.lab作为SAN条目但没有IP地址，则将“TLS验证模式”设置为“开”的发现成功，但来自MRA客户端的实际通信可能会以其他FQDN或IP为目标，因此不会通过TLS验证。

X14.2.0及更高版本

从X14.2.0版本开始，Expressway服务器会对通过流量服务器发出的每个HTTPS请求执行TLS证书

验证。这意味着在发现CUCM/IM&P/Unity节点期间，当“TLS验证模式”设置为“关闭”时，它也会执行此功能。如果验证失败，则TLS握手不会完成，请求也会失败，这可能会导致功能丢失，例如冗余或故障切换问题或完全登录失败。此外，如果“TLS验证模式”设置为“开”，则不保证所有连接都能正常工作，如后面的示例所述。

Expressway向CUCM/IM&P/Unity节点检查的确切证书如MRA指南部分所示。

除了默认的TLS验证，X14.2中还引入了一个更改，该更改可能通告密码列表的不同首选项顺序，具体取决于您的升级路径。这可能会导致在软件升级后出现意外的TLS连接，因为在升级之前，它可能从CUCM（或任何具有用于ECDSA算法的单独证书的其他产品）请求Cisco Tomcat或Cisco CallManager证书，但在升级之后，它请求ECDSA变体（实际上比RSA更安全的密码变体）。Cisco Tomcat-ECDSA或Cisco CallManager-ECDSA证书可以由其他CA签署，也可以仅由自签名证书签署（默认）。

此密码首选项顺序更改并不总是与您相关，因为它取决于升级路径，如Expressway X14.2.1 发行版本注释中所示。简而言之，您可以从维护>安全>加密器中查看每个加密列表是否预置“ECDHE-RSA-AES256-GCM-SHA384：”。如果没有，则首选较新的ECDSA密码而不是RSA密码。如果是，则您有与以前一样的行为，RSA具有更高的优先级。

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**).

在此场景中，TLS验证失败有两种方式，将在后面详细介绍：

1. 签署远程证书的CA不受信任

a.自签名证书

b.由未知CA签名的证书

2. 证书中不包含连接地址（FQDN或IP）

故障排除场景

接下来的场景显示了实验室环境中的类似场景，其中Expressway从X14.0.7升级到X14.2后，MRA登录失败。它们在日志中共享相似之处，但分辨率不同。这些日志只由在MRA登录之前启动并在MRA登录失败之后停止的诊断日志记录(从维护>诊断>诊断日志记录中)收集。没有为其启用其他调试日志记录。

1. 签署远程证书的CA不受信任

远程证书可由未包含在Expressway-C的信任库中的CA签署，也可以是未添加到Expressway-C服务器的信任库中的自签名证书（本质上也为CA）。

在下面的示例中，您可以看到指向CUCM (10.48.36.215 - cucm.steven.lab)的请求已在端口8443上

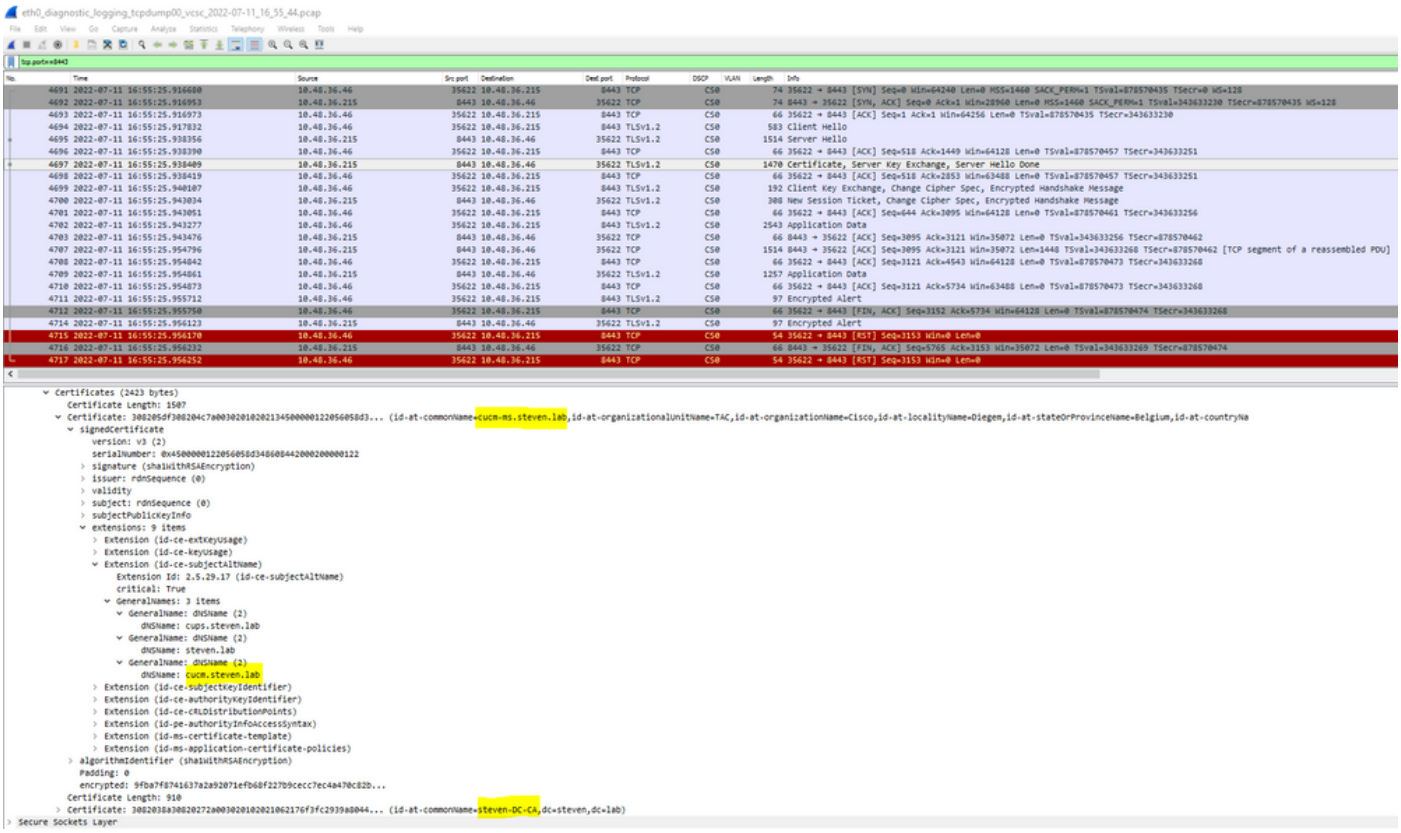
正确处理（200 OK响应），但是它会在端口6972上针对TFTP连接引发错误（502响应）。

<#root>

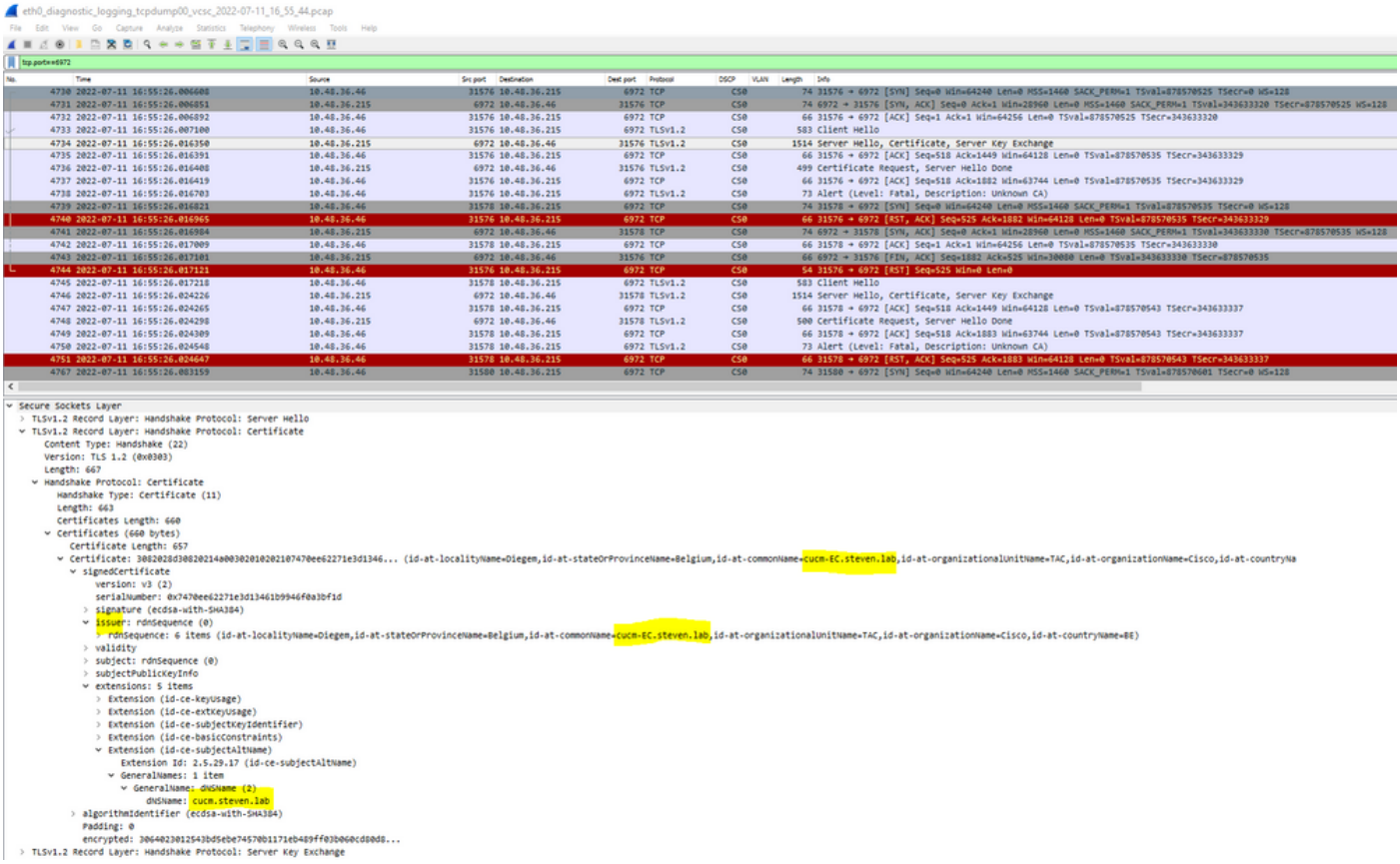
```
===Success connection on 8443===

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

2022-07-11T18:55:25.917+02:00 vcsc traffic_server[18242]:  Event="Request Allowed" Detail="Access allowe
2022-07-11T18:55:25.917+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,916" Module="netw
2022-07-11T18:55:25.955+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,955" Module="netw
2022-07-11T18:55:25.956+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:25,955" Module="netw

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="netw

2022-07-11T18:55:26.006+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:26,006" Module="netw
2022-07-11T18:55:26.016+02:00 vcsc traffic_server[18242]: UTCTime="2022-07-11 16:55:26,016" Module="netw
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="netw

502 connect failed

"
```

“证书验证失败”的错误表明Expressway-C无法验证TLS握手。原因显示在警告行中，因为它表示自签名证书。如果深度显示为0，则为自签名证书。当深度大于0时，意味着它具有证书链，因此由未知CA签名（从Expressway-C的角度来看）。

当我们查看从文本日志中提到的时间戳收集的pcap文件时，您可以看到CUCM向8443端口上的Expressway-C提供证书，其中CN为cucm-ms.steven.lab（SAN为cucm.steven.lab），STEVEN-DC-CA为cucm.steven.lab。

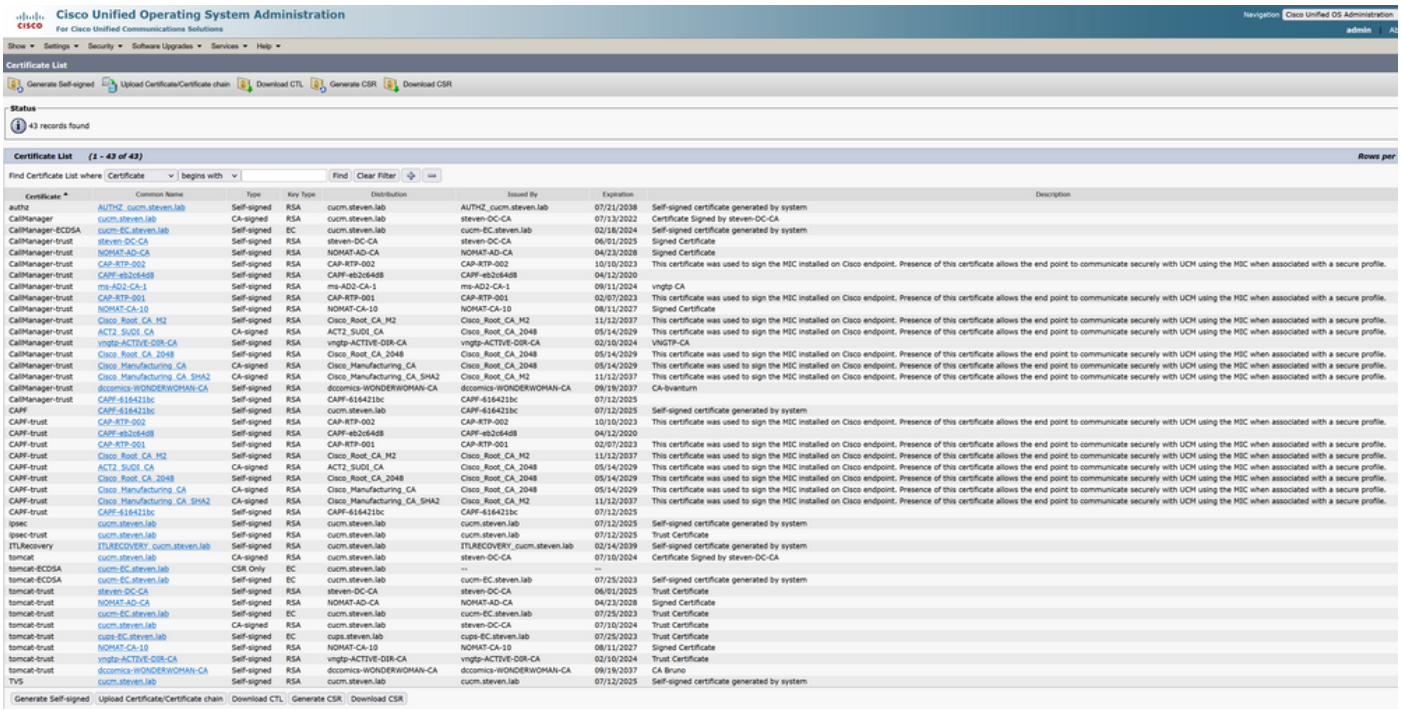


但是，当我们检查端口6972上提供的证书时，您可以看到它是自签名证书（颁发者自身），其中CN设置为cucm-EC.steven.lab。-EC扩展表明这是CUCM上设置的ECDSA证书。



在Cisco Unified OS Administration下的CUCM上，可以在Security > Certificate Management下查看就地证书，如以下示例所示。它显示tomcat和tomcat-ECDSA的不同证书，其中tomcat是CA签名的

（并受Expressway-C信任），而tomcat-ECDSA证书是自签名的，不受Expressway-C信任。



2. 证书中不包含连接地址（FQDN或IP）

除了信任库外，流量服务器还验证MRA客户端向哪个连接地址发出请求。例如，当您在CUCM上的System > Server下设置CUCM时，CUCM上的IP地址(10.48.36.215)会向Expressway-C发送此类通告，然后来自客户端（通过Expressway-C代理）的后续请求将针对此地址。

如果该特定连接地址未包含在服务器证书中，则TLS验证也会失败，并引发502错误，从而导致MRA登录失败。

<#root>

```
2022-07-11T19:49:01.472+02:00 vcsc traffic_server[3916]: UTCTime="2022-07-11 17:49:01,472" Module="netw
HTTPMSG:
|GET http://vcs_control.steven.lab:8443/c3RldmVuLmxhYi9odHRwcy8xMC40OC4zNi4yMTUvODQ0Mw/cucm-uds/user/em
...

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

其中，c3RldmVuLmxhYi9odHRwcy8xMC40OC4zNi4yMTUvODQ0Mw将(base64)转换为steven.lab/https/10.48.36.215/8443，这表示它必须将指向10.48.36.215的连接作为连接地址，而不是转换为cucm.steven.lab。如数据包捕获所示，CUCM tomcat证书不包含SAN中的IP地址，因此引发错误。

如何轻松验证

通过接下来的步骤，您可以验证您是否能够轻松实现此行为更改：

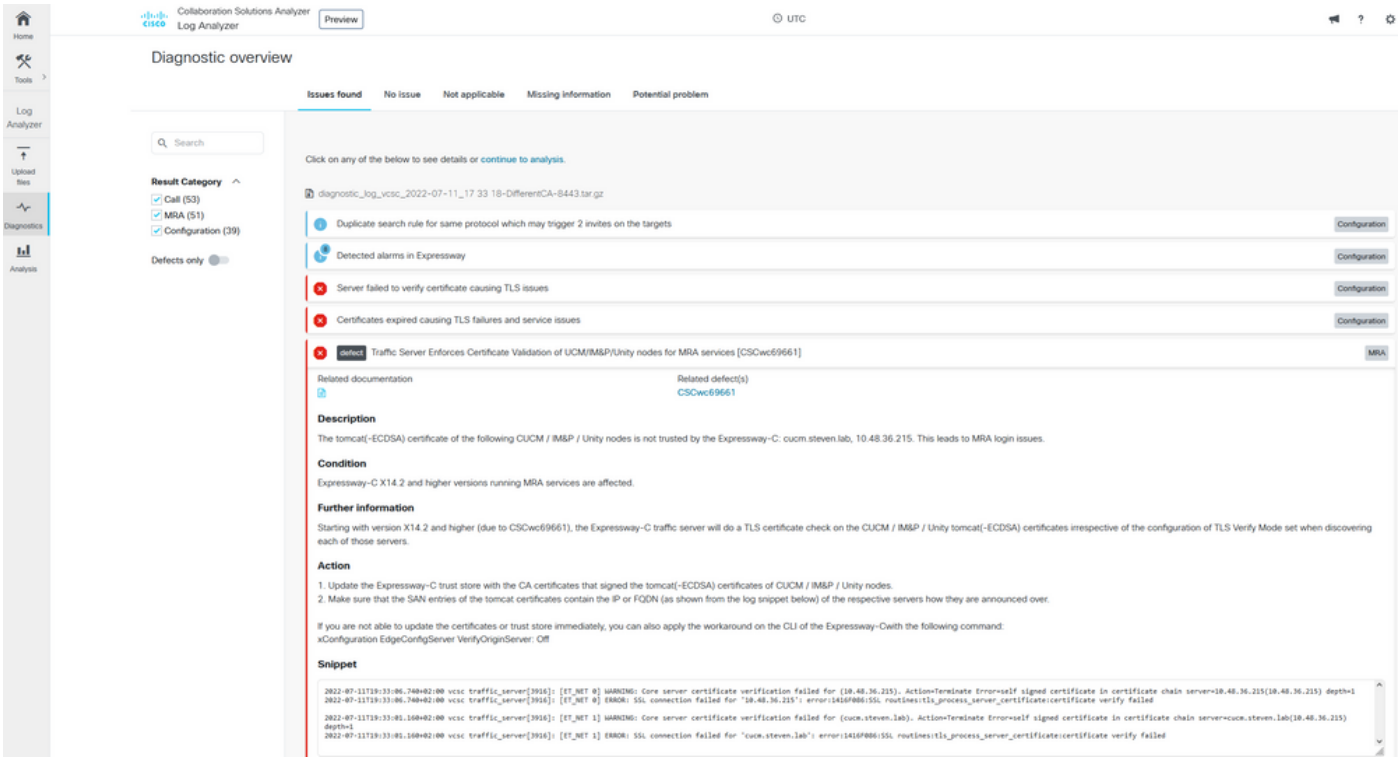
1. 在Expressway E和Expressway C服务器上启动诊断日志记录（最好启用TCPDumps），方法是选择维护>诊断>诊断日志记录（如果是集群，则从主节点启动即可）

2. 尝试MRA登录或测试升级后中断的功能

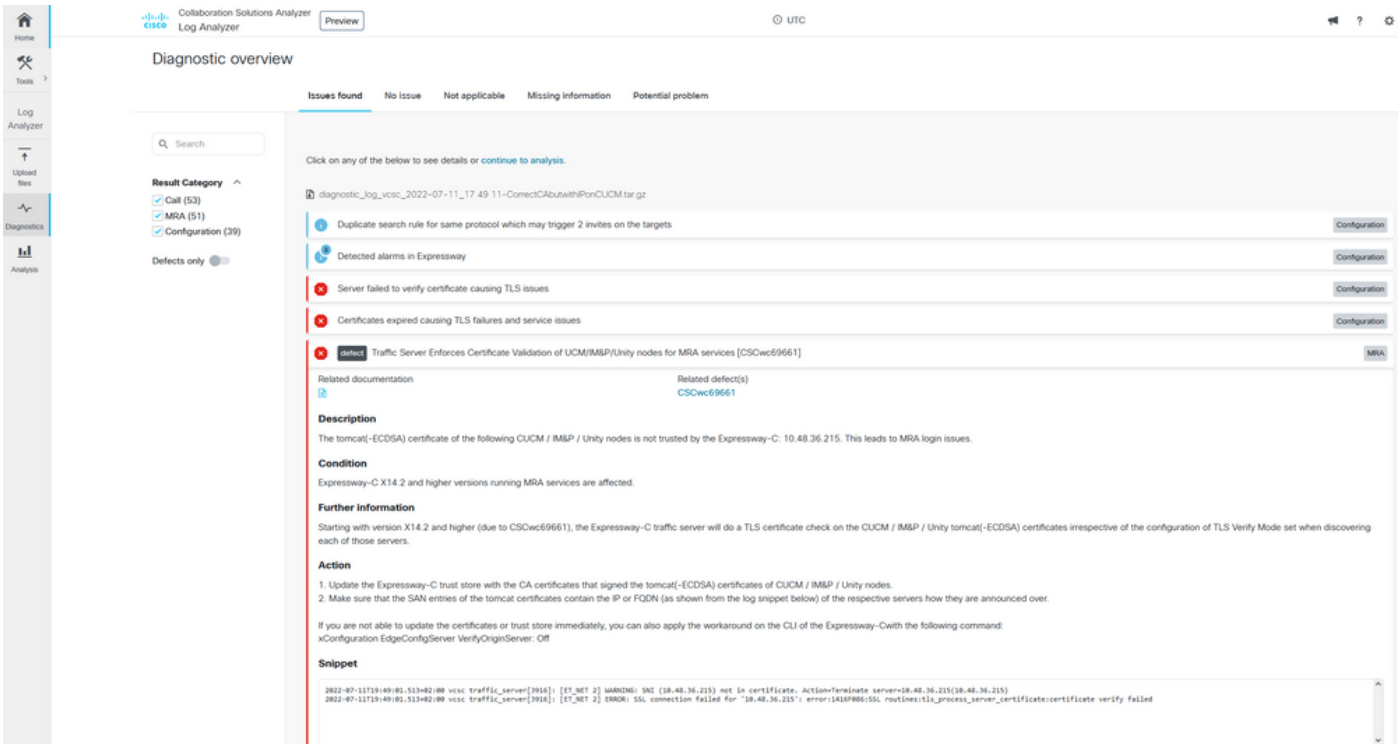
3. 等到失败，然后停止Expressway-E和Expressway-C服务器上的诊断日志记录（如果是集群，请确保分别从集群的每个节点收集日志）

4. 上传并分析协作[解决方案分析器工具]上的日志

5. 如果遇到问题，它会为每个受影响的服务器选取与此更改相关的最新警告和错误行



CA诊断签名



SNI诊断签名

解决方案

长期解决方案是确保TLS验证正常工作。要执行的操作取决于显示的警告消息。

当您看到WARNING： Core server certificate verification failed for (<server-FQDN-or-IP>)时。Action=Terminate Error=self signed certificate server=cucm.steven.lab(10.48.36.215) depth=x消息，然后您需要相应地更新Expressway-C服务器上的信任库。使用CA链签名此证书（深度> 0），或使用维护>安全>受信任CA证书中的自签名证书（深度= 0）。确保在群集中的每个服务器上执行此操作。另一种方案是，由Expressway-C信任库上的已知CA对远程证书进行签名。

注意：Expressway不允许将两个不同（例如自签名）的证书上传到Expressway的信任库中，这些证书根据思科漏洞ID CSCwa12905具有相同的公用名(CN)。要更正此问题，请转到CA签名的证书或将您的CUCM升级到版本14，您可以在其中对Tomcat和CallManager重复使用同一（自签名）证书。

如果看到WARNING： SNI (<server-FQDN-or-IP>) not in certificate消息，则表明此服务器FQDN或IP未包含在已提供的证书中。您可以修改证书以包含该信息，也可以修改配置（例如，在CUCM上的System > Server上修改为服务器证书中包含的内容），然后刷新Expressway-C服务器上的配置以考虑该配置。

相关信息

短期解决方案是应用所记录的解决方法，以回退到X14.2.0之前的运行模式。您可以通过Expressway-C服务器节点上的CLI使用新引入的命令对此执行操作：

```
xConfiguration EdgeConfigServer VerifyOriginServer: Off
```

It

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。