了解默认情况下的CUCM安全性以及ITL运行和故障排除

目录

简介

背景信息

SBD概述

- TFTP下载身份验证
- TFTP配置文件加密
- 信任验证服务（远程证书和签名验证）

SBD详细信息和故障排除信息

- CUCM上存在的ITL文件和证书
- 电话下载ITL和配置文件
- 电话验证ITL和配置文件
- 未知证书的电话联系人TVS
- 手动验证电话ITL是否与CUCM ITL匹配

限制和交互

- 重新生成证书/重建集群/证书过期
- 在集群之间移动电话
- 备份和恢复
- 更改主机名或域名
- 集中式TFTP

常见问题解答

- 我可以关闭SBD吗？
- CallManager.pem丢失后，能否从所有电话轻松删除ITL文件？

简介

本文档介绍Cisco Unified Communications Manager(CUCM)版本8.0及更高版本的默认安全(SBD)功能。

背景信息

CUCM 8.0及更高版本引入了SBD功能，包括身份信任列表(ITL)文件和信任验证服务(TVS)。

每个CUCM集群现在自动使用基于ITL的安全性。在安全性与易用性/易于管理性之间有一个折衷，管理员在对8.0版CUCM群集进行某些更改之前必须了解这一点。

本文档是对官方默认安全性文档的补充，提供了操作信息和故障排除提示，可帮助管理员简化故障排除过程。

熟悉SBD的这些核心概念是一个好主意：非对称密钥加密 Wikipedia 文章 和Public Key Infrastructure Wikipedia 文章 。

SBD概述

本部分简要概述了SBD提供的具体功能。有关每项功能的完整技术详细信息，请参见SBD详细信息和故障排除信息部分。

SBD为支持的IP电话提供以下三种功能：

- 使用签名密钥的TFTP下载文件（配置、区域设置、铃声）的默认身份验证
- 使用签名密钥的TFTP配置文件的可选加密
- 使用CUCM(TVS)上的远程证书信任存储区的电话发起HTTPS连接的证书验证

本文档概述了其中的每个功能。

TFTP下载身份验证

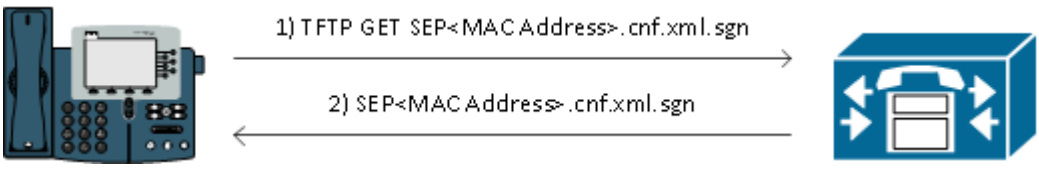
当存在证书信任列表(CTL)或ITL文件时，IP电话将从CUCM TFTP服务器请求已签名的TFTP配置文件。

此文件允许电话验证配置文件是否来自受信任的来源。如果电话上存在CTL/ITL文件，则配置文件必须由可信TFTP服务器签名。

文件在传输时是网络中的纯文本，但带有特殊的验证签名。

电话请求SEP<MAC Address>.cnf.xml.sgn以接收具有特殊签名的配置文件。

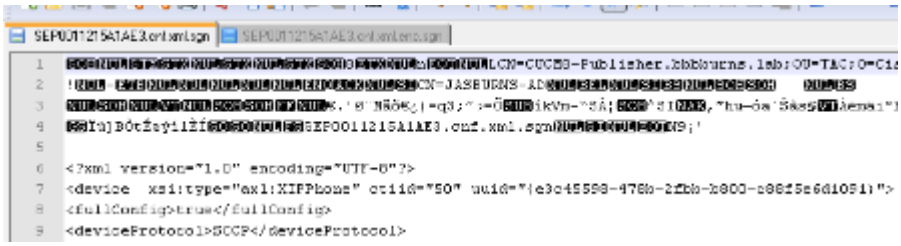
此配置文件由与操作系统(OS)管理证书管理(Operating System [OS] Administration Certificate Management)页面上的CallManager.pem对应的TFTP私钥签名。



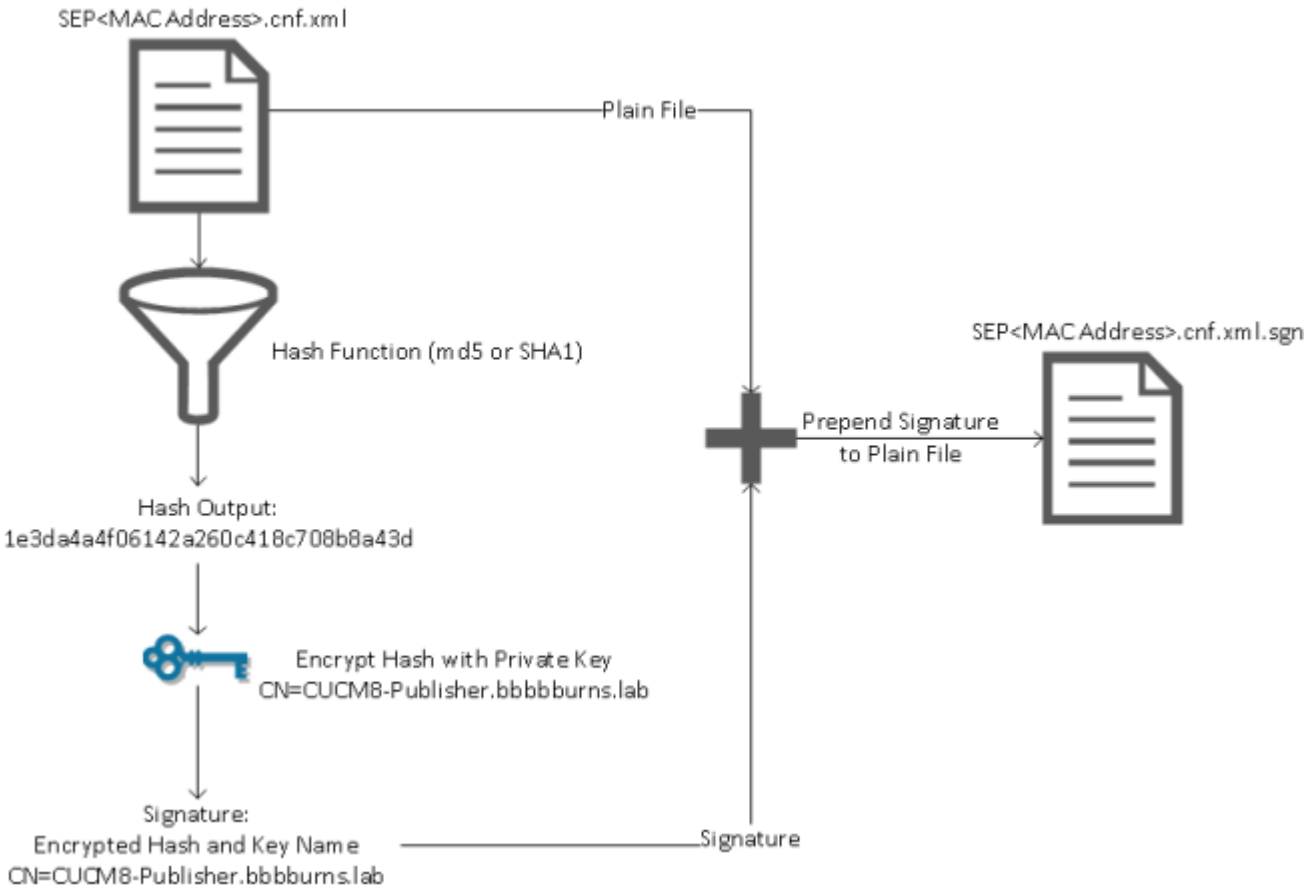
签名文件在顶部有一个签名以对文件进行身份验证，否则以纯文本XML显示。

下图显示配置文件的签名者是CN=CUCM8-Publisher.bbbburns.lab，后者依次由CN=JASBURNS-AD签名。

这意味着在接受此配置文件之前，电话需要根据ITL文件验证CUCM8-Publisher.bbburns.lab的签名。

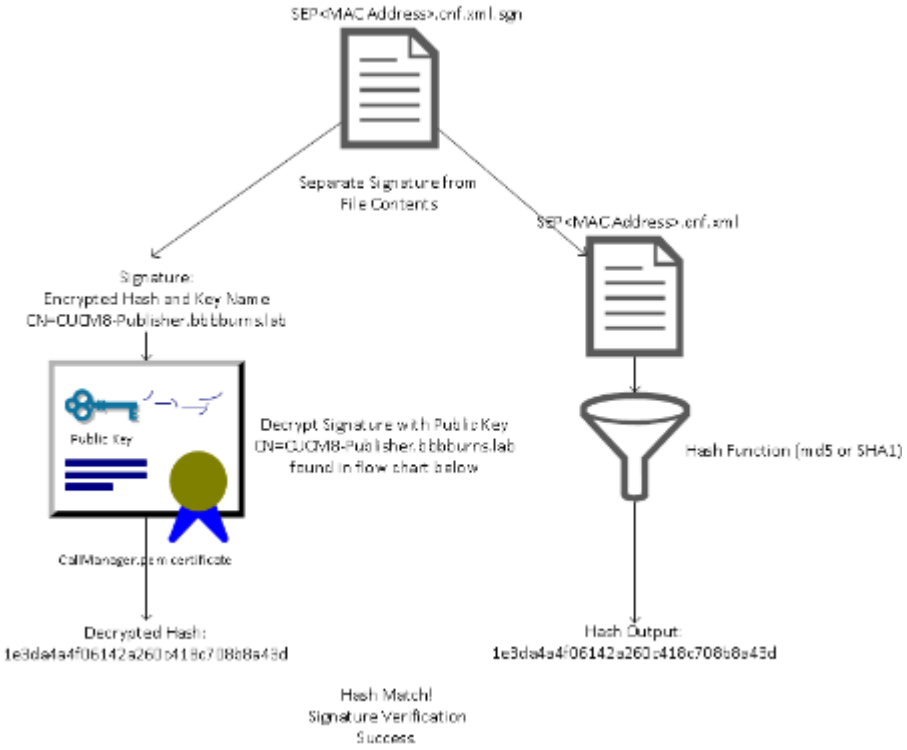


下图显示了如何使用私钥以及消息摘要算法(MD)5或安全散列算法(SHA)1散列函数来创建签名文件。



签名验证通过使用匹配的公钥来解密散列，从而逆转此过程。如果散列匹配，则显示：

- 在传输过程中未修改此文件。
- 此文件来自签名中列出的参与方，因为使用公钥成功解密的任何内容都必须使用私钥进行加密。



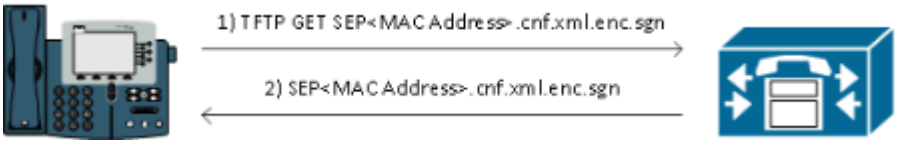
TFTP配置文件加密

如果在关联的电话安全配置文件中启用了可选的TFTP配置加密，则电话会请求加密配置文件。

此文件使用TFTP私钥签名，并使用电话和CUCM之间交换的对称密钥进行加密(有关详细信息，请参阅Cisco Unified Communications Manager安全指南8.5(1)版本)。

网络嗅探器无法读取其内容，除非观察者具有必要的密钥。

电话请求SEP<MAC Address>.cnf.xml.enc.sgn以获取已签名的加密文件。



加密的配置文件在开头也有签名，但之后没有纯文本数据，只有加密数据（在此文本编辑器中被损坏的二进制字符）。

该图显示签名人与上一个示例相同，因此此签名人必须出现在ITL文件中，电话才能接受该文件。

此外，解密密钥必须正确无误，电话才能读取文件的内容。



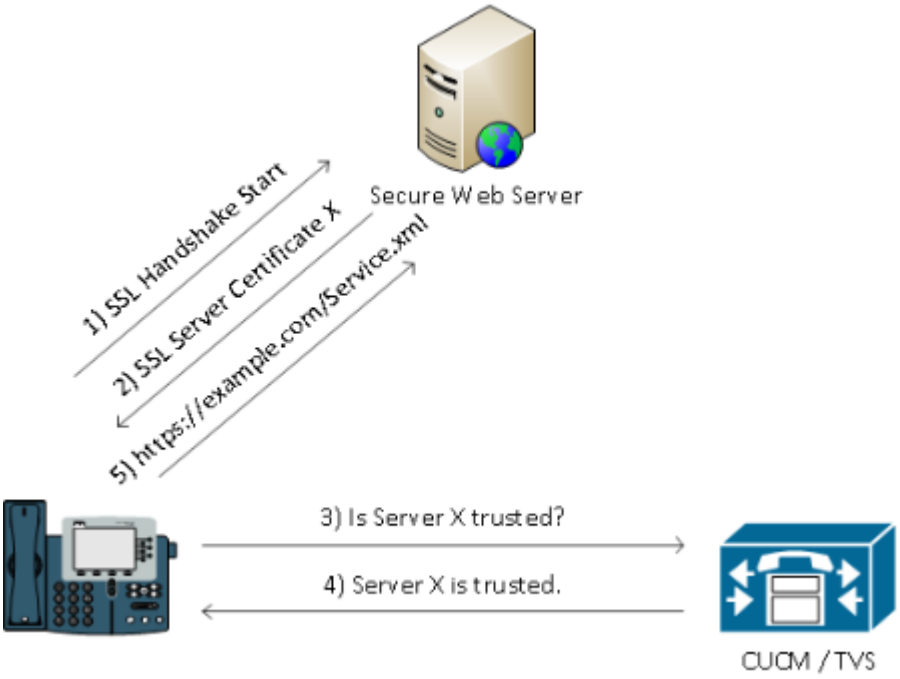
信任验证服务（远程证书和签名验证）

IP电话包含有限的内存，而且网络中还可以管理大量电话。

CUCM通过TVS充当远程信任库，因此无需在每个IP电话上放置完整的证书信任库。

每当电话无法通过CTL或ITL文件验证签名或证书时，它会要求TVS服务器进行验证。

与信任存储存在于所有IP电话上相比，此中心信任存储更易于管理。



SBD详细信息和故障排除信息

本节详细介绍SBD过程。

CUCM上存在的ITL文件和证书

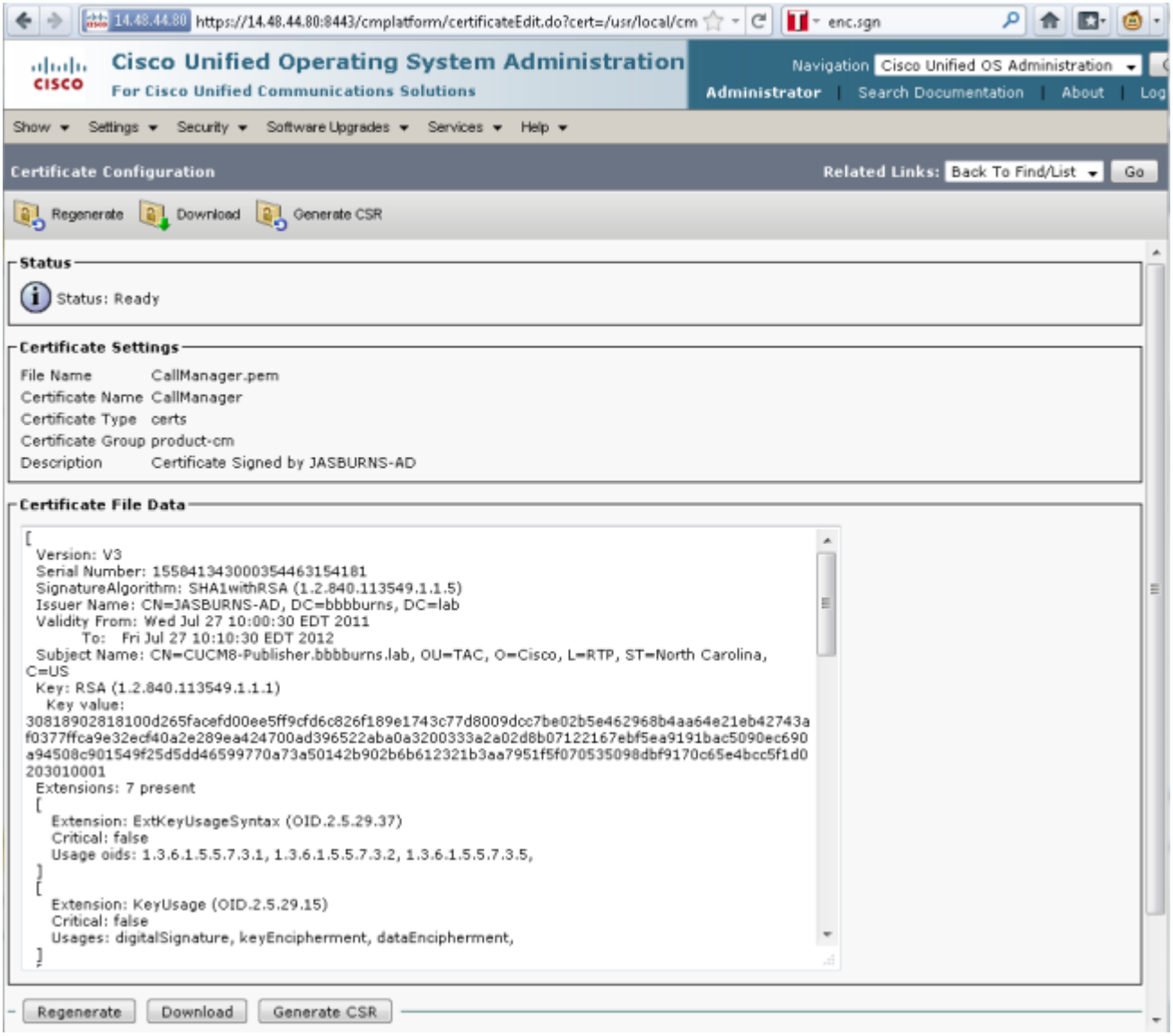
首先，CUCM服务器本身必须存在许多文件。最重要的部分是TFTP证书和TFTP私钥。

TFTP证书位于OS Administration > Security > Certificate Management > CallManager.pem下。

CUCM服务器将CallManager.pem证书私钥和公钥用于TFTP服务(以及Cisco Call Manager(CCM)服务)。

该图显示CallManager.pem证书颁发给CUCM8-publisher.bbbburns.lab,并由JASBURNS-AD签名。所有TFTP配置文件均使用以下私钥进行签名。

所有电话都可以使用CallManager.pem证书中的TFTP公钥解密任何使用TFTP私钥加密的文件，以及验证任何使用TFTP私钥签名的文件。



除CallManager.pem证书私钥外，CUCM服务器还存储提供给电话的ITL文件。

show itl命令通过对CUCM服务器操作系统CLI的安全外壳(SSH)访问显示此ITL文件的完整内容。

本节逐一分解国际交易日志文件，因为它包含电话使用的许多重要组件。

第一部分是签名信息。甚至国际交易日志文件也是已签名的文件。此输出显示其签名的是与之前的CallManager.pem证书关联的TFTP私钥。

<#root>

```
admin:
show itl


Length of ITL file: 5438
The ITL File was last modified on Wed Jul 27 10:16:24 EDT 2011

        Parse ITL File
        ----------------
```

```
Version:        1.2
HeaderLength:   296 (BYTES)

BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
3       SIGNERID        2       110
4       SIGNERNAME      76      CN=CUCM8-Publisher.bbbburns.lab;
                                OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
5       SERIALNUMBER    10      21:00:2D:17:00:00:00:00:00:05
6       CANAME          15      CN=JASBURNS-AD
*Signature omitted for brevity*
```

接下来的每个部分都包含其在一个特殊Function参数中的用途。第一个功能是系统管理员安全令牌。这是TFTP公钥的签名。

```
        ITL Record #:1
                  ----
BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
1       RECORDLENGTH    2       1972
2       DNSNAME         2
3       SUBJECTNAME     76      CN=CUCM8-Publisher.bbbburns.lab;
                                OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
4       FUNCTION        2       System Administrator Security Token
5       ISSUERNAME      15      CN=JASBURNS-AD
6       SERIALNUMBER    10      21:00:2D:17:00:00:00:00:00:05
7       PUBLICKEY       140
8       SIGNATURE       256
9       CERTIFICATE     1442    0E 1E 28 0E 5B 5D CC 7A 20 29 61 F5
                                8A DE 30 40 51 5B C4 89 (SHA1 Hash HEX)
This etoken was used to sign the ITL file.
```

下一个功能是CCM+TFTP。这也是TFTP公钥，用于验证和解密下载的TFTP配置文件。

```
        ITL Record #:2
                  ----
BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
1       RECORDLENGTH    2       1972
2       DNSNAME         2
3       SUBJECTNAME     76      CN=CUCM8-Publisher.bbbburns.lab;
                                OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
4       FUNCTION        2       CCM+TFTP
5       ISSUERNAME      15      CN=JASBURNS-AD
6       SERIALNUMBER    10      21:00:2D:17:00:00:00:00:00:05
7       PUBLICKEY       140
8       SIGNATURE       256
9       CERTIFICATE     1442    0E 1E 28 0E 5B 5D CC 7A 20 29 61 F5
                                8A DE 30 40 51 5B C4 89 (SHA1 Hash HEX)
```

下一个功能是TVS。电话连接的每个TVS服务器的公钥都有一个条目。

这样，电话就可以建立到TVS服务器的安全套接字层(SSL)会话。

```
        ITL Record #:3
                  ----
BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
1       RECORDLENGTH    2       743
2       DNSNAME         2
3       SUBJECTNAME     76      CN=CUCM8-Publisher.bbbburns.lab;
                                OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
4       FUNCTION        2       TVS
5       ISSUERNAME      76      CN=CUCM8-Publisher.bbbburns.lab;
                                OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
6       SERIALNUMBER    8       2E:3E:1A:7B:DA:A6:4D:84
7       PUBLICKEY       270
8       SIGNATURE       256
11      CERTHASH        20      C7 E1 D9 7A CC B0 2B C2 A8 B2 90 FB
                                AA FE 66 5B EC 41 42 5D
12      HASH ALGORITHM  1       SHA-1
```

ITL文件中包含的最后一个功能是证书颁发机构代理功能(CAPF)。

此证书允许电话建立到CUCM服务器上的CAPF服务的安全连接，以便电话可以安装或更新本地重要证书(LSC)。

```
        ITL Record #:4
                  ----
BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
1       RECORDLENGTH    2       455
2       DNSNAME         2
3       SUBJECTNAME     61      CN=CAPF-9c4cba7d;
                                OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
4       FUNCTION        2       CAPF
5       ISSUERNAME      61      CN=CAPF-9c4cba7d;
                                OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
6       SERIALNUMBER    8       0A:DC:6E:77:42:91:4A:53
7       PUBLICKEY       140
8       SIGNATURE       128
11      CERTHASH        20      C7 3D EA 77 94 5E 06 14 D2 90 B1
                                A1 43 7B 69 84 1D 2D 85 2E
12      HASH ALGORITHM  1       SHA-1

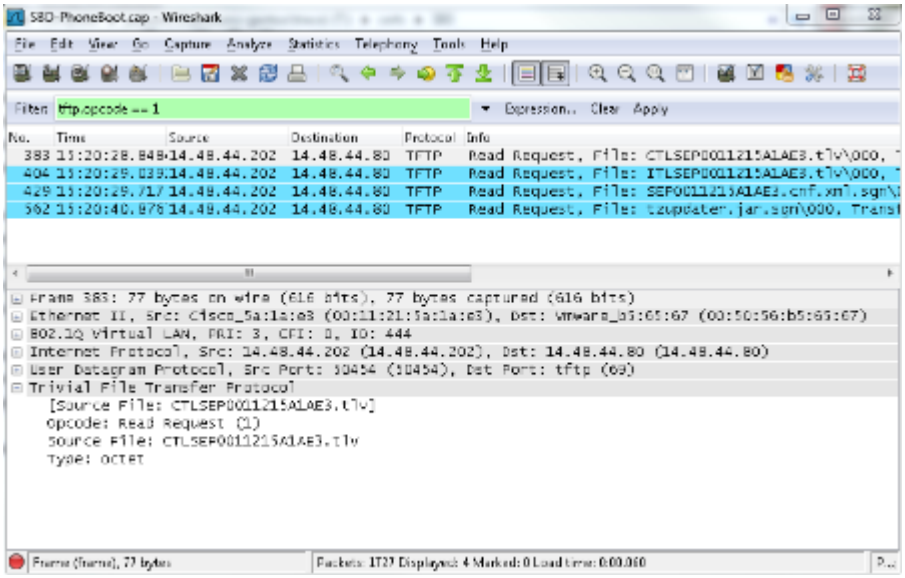
The ITL file was verified successfully.
```

下一部分将介绍电话启动时发生的具体情况。

电话下载ITL和配置文件

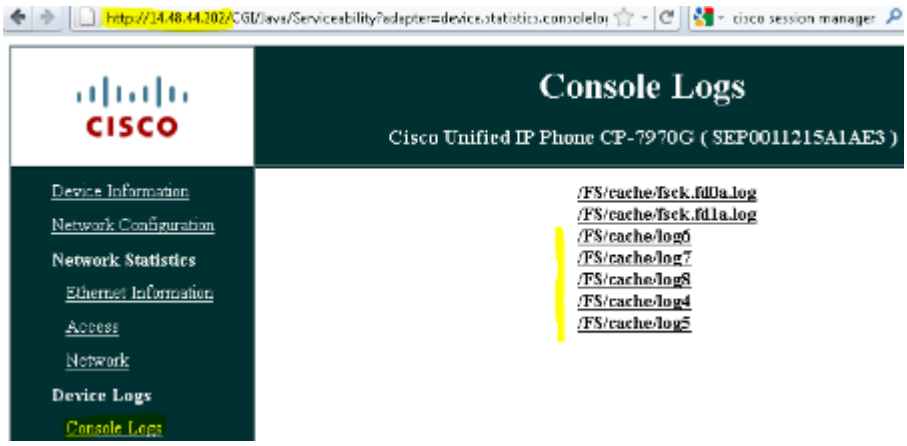
电话启动并获取IP地址和TFTP服务器地址后，首先请求CTL和ITL文件。

此数据包捕获显示对ITL文件的电话请求。如果过滤tftp.opcode == 1，您会看到来自电话的每个TFTP读取请求：



由于电话成功从TFTP接收CTL和ITL文件，因此电话会请求签名配置文件。

可以从电话Web界面获取显示此行为的电话控制台日志：



首先，电话请求一个CTL文件，成功如下：

```
837: NOT 09:13:17.561856 SECD: tlRequestFile: Request CTLSEP0011215A1AE3.tlv
846: NOT 09:13:17.670439 TFTP: [27]:Requesting CTLSEP0011215A1AE3.tlv from
14 . 48 . 44 . 80
847: NOT 09:13:17.685264 TFTP: [27]:Finished --> rcvd 4762 bytes
```

接下来，电话还请求一个ITL文件：

```
868: NOT 09:13:17.860613 TFTP: [28]:Requesting ITLSEP0011215A1AE3.tlv from
14 . 48 . 44 . 80
```

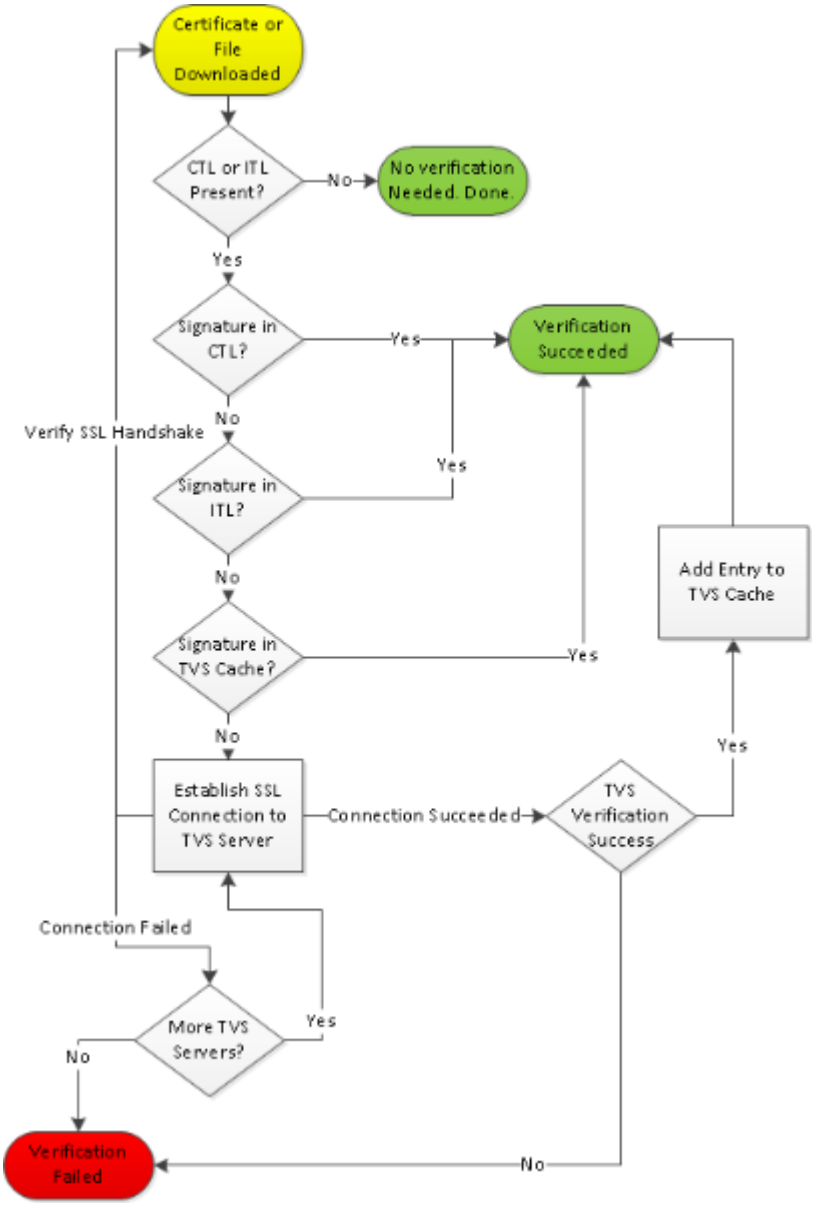
```
869: NOT 09:13:17.875059 TFTP: [28]:Finished --> rcvd 5438 bytes
```

电话验证ITL和配置文件

下载ITL文件后，必须对其进行验证。此时，电话可处于多种状态，因此本文档将介绍所有状态。

- 电话不存在CTL或ITL文件，或者由于Prepare Cluster for Rollback to Pre 8.0参数，ITL为空。在这种状态下，电话盲目信任下载的下一个CTL或ITL文件，并在以后使用此签名。
- 电话已有一个CTL，但没有ITL。在这种状态下，只有在CTL文件中的CCM+TFTP功能可以验证时，电话才信任ITL。
- 电话已经有一个CTL和一个ITL文件。在此状态下,电话会验证最近下载的文件是否与CTL、ITL或TVS服务器中的签名匹配。

下面是一个流程图，介绍电话如何验证签名文件和HTTPS证书：



在这种情况下，电话能够验证ITL和CTL文件中的签名。该电话已经具有CTL和ITL，因此只需对照它们进行检查即可找到正确的签名。

```
877: NOT 09:13:17.925249 SECD: validate_file_envelope:
File sign verify SUCCESS; header length <296>
```

由于电话下载了CTL和ITL文件，因此从这一点起，它只请求已签名的配置文件。

这说明电话逻辑是基于CTL和ITL的存在来确定TFTP服务器是否安全，然后请求签名文件：

```
917: NOT 09:13:18.433411 tftpClient: tftp request rcv'd from /usr/tmp/tftp,
srcFile = SEP0011215A1AE3.cnf.xml, dstFile = /usr/ram/SEP0011215A1AE3.cnf.xml
max size = 550001
918: NOT 09:13:18.457949 tftpClient: auth server - tftpList[0] = ::ffff:
14 . 48 . 44 . 80
919: NOT 09:13:18.458937 tftpClient: look up server - 0
920: NOT 09:13:18.462479 SECD: lookupCTL: TFTP SRVR secure
921: NOT 09:13:18.466658 tftpClient: secVal = 0x9 922: NOT 09:13:18.467762
tftpClient: ::ffff:14 . 48 . 44 . 80 is a secure server
923: NOT 09:13:18.468614 tftpClient: retval = SRVR_SECURE
924: NOT 09:13:18.469485 tftpClient: Secure file requested
925: NOT 09:13:18.471217 tftpClient: authenticated file approved - add .sgn
-- SEP0011215A1AE3.cnf.xml.sgn
926: NOT 09:13:18.540562 TFTP: [10]:Requesting SEP0011215A1AE3.cnf.xml.sgn
from 14 . 48 . 44 . 80 with size limit of 550001
927: NOT 09:13:18.559326 TFTP: [10]:Finished --> rcvd 7652 bytes
```

下载已签名的配置文件后，电话必须根据ITL中的CCM+TFTP功能对其进行身份验证：

```
937: NOT 09:13:18.656906 SECD: verifyFile: verify SUCCESS
</usr/ram/SEP0011215A1AE3.cnf.xml>
```

未知证书的电话联系人TVS

ITL文件提供一个TVS函数，其中包含在CUCM服务器TCP端口2445上运行的TVS服务的证书。

TVS在激活CallManager服务的所有服务器上运行。CUCM TFTP服务使用已配置的CallManager组在电话配置文件中建立电话必须联系的TVS服务器列表。

有些实验只使用一台CUCM服务器。在多节点CUCM集群中，一个电话最多可以有三个TVS条目，该电话的CUCM组中的每个CUCM对应一个。

此示例显示了按下IP电话上的Directories按钮后会发生什么情况。目录URL配置为HTTPS，因此电话会收到来自目录服务器的Tomcat Web证书。

此Tomcat Web证书（操作系统管理中的tomcat.pem）未加载到电话中，因此电话必须与TVS联系才能对证书进行身份验证。

有关交互的说明，请参阅之前的TVS概述图。以下是电话控制台日志的视角：

首先找到目录URL:

```
1184: NOT 15:20:55.219275 JVM: Startup Module Loader|cip.dir.TandunDirectories:
? - Directory url https://14 . 48 . 44 . 80:8443/ccmcip/xmldirectory.jsp
```

这是需要验证的SSL/传输层安全(TLS)安全HTTP会话。

```
1205: NOT 15:20:59.404971 SECD: clpSetupSsl: Trying to connect to IPV4, IP:
14 . 48 . 44 . 80, Port : 8443
1206: NOT 15:20:59.406896 SECD: clpSetupSsl: TCP connect() waiting,
<14 . 48 . 44 . 80> c:8 s:9 port: 8443
1207: NOT 15:20:59.408136 SECD: clpSetupSsl: TCP connected,
<14 . 48 . 44 . 80> c:8 s:9
1208: NOT 15:20:59.409393 SECD: clpSetupSsl: start SSL/TLS handshake,
<14 . 48 . 44 . 80> c:8 s:9
1209: NOT 15:20:59.423386 SECD: srvr_cert_vfy: Server Certificate
Validation needs to be done
```

电话首先验证SSL/TLS服务器提供的证书是否在CTL中。然后，电话查看ITL文件中的函数，以查看是否找到匹配项。

此错误消息显示“HTTPS证书不在CTL中”，这意味着“在CTL或ITL中找不到证书”。

```
1213: NOT 15:20:59.429176 SECD: findByCertAndRoleInTL: Searching TL from CTL file
1214: NOT 15:20:59.430315 SECD: findByCertAndRoleInTL: Searching TL from ITL file
1215: ERR 15:20:59.431314 SECD: EROR:https_cert_vfy: HTTPS cert not in CTL,
<14 . 48 . 44 . 80>
```

在检查CTL和ITL文件的直接内容以获取证书后，电话接下来检查的是TVS缓存。

如果电话最近向TVS服务器请求了相同的证书，则这样做是为了减少网络流量。

如果在电话缓存中找不到HTTPS证书，您可以与TVS服务器本身建立TCP连接。

```
1220: NOT 15:20:59.444517 SECD: processTvsClntReq: TVS Certificate
Authentication request
1221: NOT 15:20:59.445507 SECD: lookupAuthCertTvsCacheEntry: No matching
entry found at cache
1222: NOT 15:20:59.446518 SECD: processTvsClntReq: No server sock exists,
must be created
1223: NOT 15:20:59.451378 SECD: secReq_initClient: clnt sock fd 11 bound
to </tmp/secClnt_secd>
```

```
1224: NOT 15:20:59.457643 SECD: getTvsServerInfo: Phone in IPv4 only mode
1225: NOT 15:20:59.458706 SECD: getTvsServerInfo: Retreiving IPv4 address
1230: NOT 15:20:59.472628 SECD: connectToTvsServer: Successfully started
a TLS connection establishment to the TVS server: IP:14 . 48 . 44 . 80, port:2445
(default); Waiting for it to get connected.
```

请记住，与TVS本身的连接是SSL/TLS（安全HTTP或HTTPS），因此它也是需要根据CTL到ITL进行身份验证的证书。

如果一切正常，TVS服务器证书可在ITL文件的TVS功能中找到。参见上例ITL文件中的ITL记录#3。

```
1244: NOT 15:20:59.529938 SECD: srvr_cert_vfy: Server Certificate Validation
needs to be done
1245: NOT 15:20:59.533412 SECD: findByIssuerAndSerialAndRoleInTL:
Searching TL from CTL file
1246: NOT 15:20:59.534936 SECD: findByIssuerAndSerialAndRoleInTL:
Searching TL from ITL file
1247: NOT 15:20:59.537359 SECD: verifyCertWithHashFromTL: cert hash and
hash in TL MATCH
1248: NOT 15:20:59.538726 SECD: tvs_cert_vfy: TVS cert verified with hash
from TL, <14 . 48 . 44 . 80>
```

成功！电话现在与TVS服务器具有安全连接。下一步是询问TVS服务器“Hello， Do I trust this Directories server certificate？”

此示例显示了该问题的答案 — 值为0的响应，表示成功（无错误）。

```
1264: NOT 15:20:59.789738 SECD: sendTvsClientReqToSrvr: Authenticate
Certificate : request sent to TVS server - waiting for response
1273: NOT 15:20:59.825648 SECD: processTvsSrvrResponse: Authentication Response
received, status : 0
```

由于TVS的响应成功，因此该证书的结果会保存到缓存中。

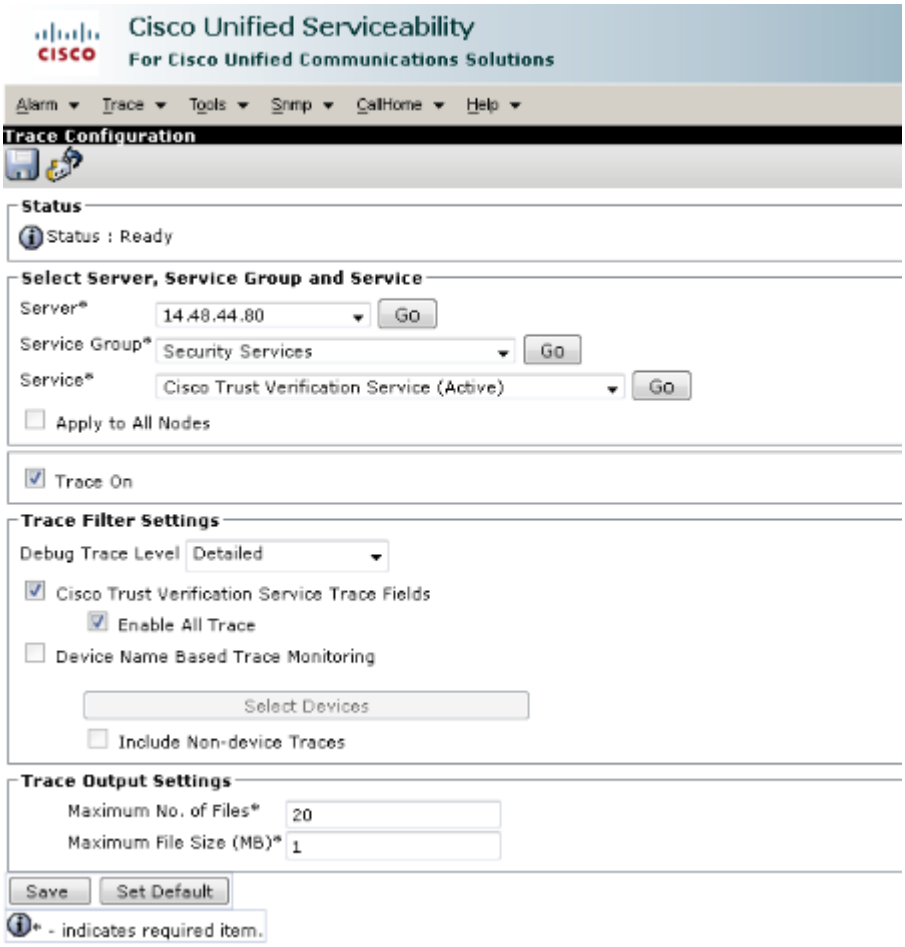
这意味着，如果您在接下来的86,400秒内再次按Directories按钮，则不需要联系TVS服务器以验证证书。您只需访问本地缓存。

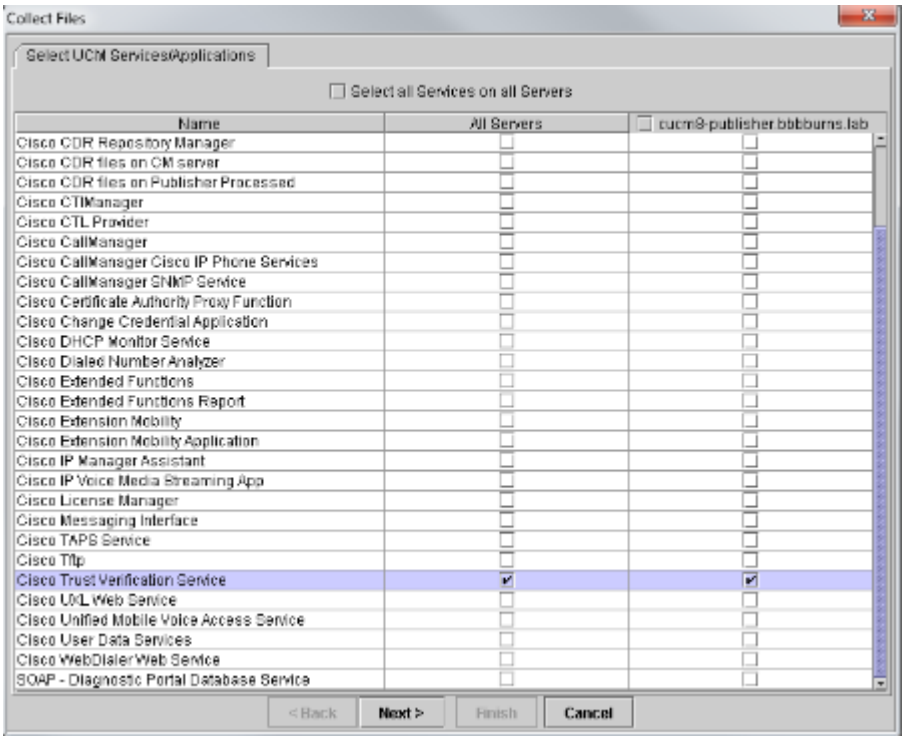
```
1279: NOT 15:20:59.837086 SECD: saveCertToTvsCache: Saving certificate
in TVS cache with default time-to-live value: 86400 seconds
1287: ERR 15:20:59.859993 SECD: Authenticated the HTTPS conn via TVS
```

最后，您验证您与目录服务器的连接是否成功。

```
1302: ERR 15:21:01.959700 JVM: Startup Module Loader|cip.http.ae:?
- listener.httpSucceed:  https://14 . 48 . 44 . 80:8443/ccmcip/
xmldirectoryinput.jsp?name=SEP0011215A1AE3
```

以下是TVS运行的CUCM服务器上发生情况的示例。您可以使用思科统一实时监控工具(RTMT)收集TVS日志。





CUCM TVS日志显示您与电话的SSL握手，电话询问TVS有关Tomcat证书，然后TVS响应指示证书在TVS证书存储区匹配。

```
15:21:01.954 |   debug 14 . 48 . 44 . 202: tvsSSLHandShake Session ciphers - AES256-SHA
15:21:01.954 |   debug TLS HS Done for ph_conn .
15:21:02.010 |   debug     MsgType               : TVS_MSG_CERT_VERIFICATION_REQ
15:21:02.011 |   debug tvsGetIssuerNameFromX509 - issuerName : CN=CUCM8-
Publisher.bbbburns.lab;OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US and Length: 75

15:21:02.011 |   debug CertificateDBCache::getCertificateInformation -
Certificate compare return =0
15:21:02.011 |   debug CertificateDBCache::getCertificateInformation -
Certificate found and equal
15:21:02.011 |   debug     MsgType               : TVS_MSG_CERT_VERIFICATION_RES
```

TVS证书存储是OS Administration > Certificate Management网页中包含的所有证书的列表。

手动验证电话ITL是否与CUCM ITL匹配

故障排除中发现的一个常见误解是倾向于删除ITL文件，以期解决文件验证问题。

有时需要删除ITL文件，但只有在满足所有这些条件时才需要删除ITL文件。

- 电话上的ITL文件的签名与CM TFTP服务器上的ITL文件的签名不匹配。
- ITL文件中的TVS签名与TVS提供的证书不匹配。
- 当电话尝试下载ITL文件或配置文件时，显示“验证失败”。
- 不存在旧TFTP私钥的备份。

以下是检查前两个条件的方法。

首先，您可以将CUCM上存在的ITL文件的校验和与电话的ITL文件的校验和进行比较。

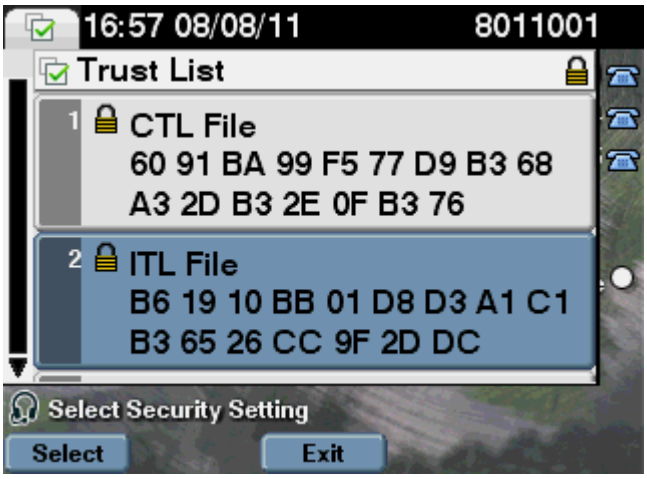
目前，无法从CUCM本身查看CUCM上ITL文件的MD5sum，除非您运行带有此Cisco bug ID CSCto60209的修复程序的版本。

在过渡期间，请使用您喜欢的GUI或CLI程序运行此工具：

```
jasburns@jasburns-gentoo /data/trace/jasburns/certs/SBD $ tftp 14 . 48 . 44 . 80
tftp> get ITLSEP0011215A1AE3.tlv
Received 5438 bytes in 0.0 seconds
tftp> quit
jasburns@jasburns-gentoo /data/trace/jasburns/certs/SBD $ md5sum
ITLSEP0011215A1AE3.tlv
b61910bb01d8d3a1c1b36526cc9f2ddc  ITLSEP0011215A1AE3.tlv
```

这表明CUCM中ITL文件的MD5sum为b61910bb01d8d3a1c1b36526cc9f2ddc。

现在，您可以查看电话本身，以确定加载到该电话的ITL文件的散列：Settings > Security Configuration > Trust List。

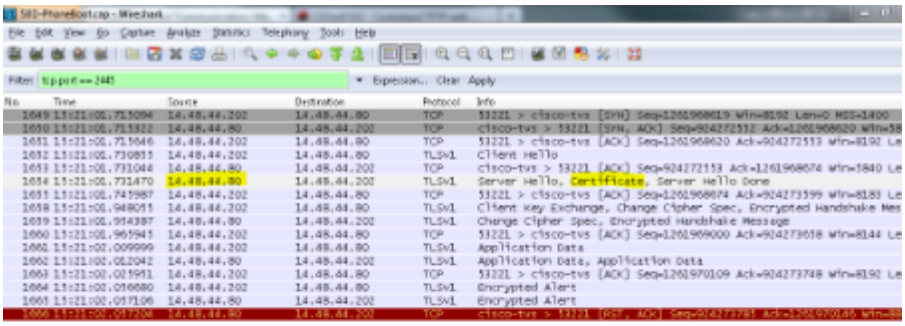


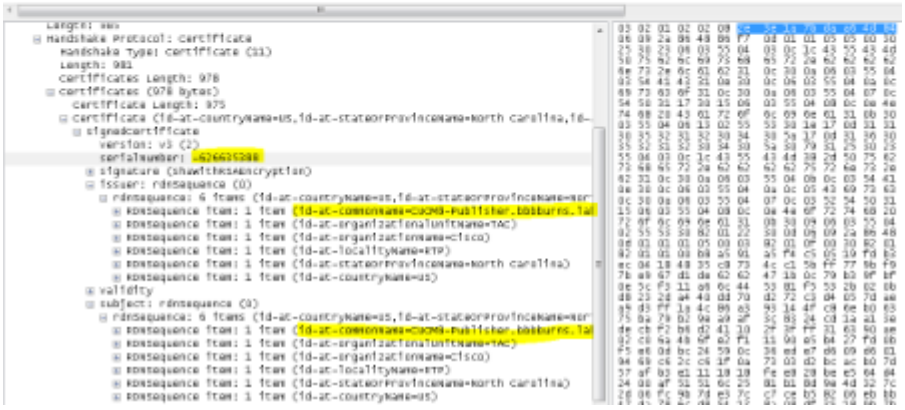
这表明MD5和匹配。这意味着电话上的ITL文件与CUCM上的文件匹配，因此不需要删除该文件。

如果匹配，您需要继续下一个操作 — 确定ITL中的TVS证书是否与TVS提供的证书匹配。此操作涉及的范围更广。

首先，查看连接到TCP端口2445上TVS服务器的电话的数据包捕获。

在Wireshark中右键单击此数据流中的任何数据包，单击Decode As，然后选择SSL。查找如下所示的服务器证书：





查看上一个ITL文件中包含的TVS证书。然后您会看到序列号为2E3E1A7BDAA64D84的条目。

<#root>

admin:

show itl

```
         ITL Record #:3
                    ----
BYTEPOS TAG              LENGTH  VALUE
------- ---              ------  -----
1       RECORDLENGTH     2       743
2       DNSNAME          2
3       SUBJECTNAME      76      CN=CUCM8-Publisher.bbbburns.lab;
                                 OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
4       FUNCTION         2       TVS
5       ISSUERNAME       76      CN=CUCM8-Publisher.bbbburns.lab;
                                 OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
6       SERIALNUMBER     8       2E:3E:1A:7B:DA:A6:4D:84
```

成功，ITL文件中的TVS.pem与网络上提供的TVS证书匹配。您无需删除ITL，并且TVS提供正确的证书。

如果文件身份验证仍失败，请检查前面的流程图的其余部分。

限制和交互

重新生成证书/重建集群/证书过期

现在最重要的证书是CallManager.pem证书。此证书私钥用于签署所有TFTP配置文件，其中包括ITL文件。

如果重新生成CallManager.pem文件，则会使用新私钥生成新的CCM+TFTP证书。此外，ITL文件现在使用这个新的CCM+TFTP密钥进行签名。

重新生成CallManager.pem并重新启动TVS和TFTP服务后，电话启动时将发生这种情况。

1. 电话尝试从TFTP服务器下载新的CCM+TFTP签署的新ITL文件。此时，电话只有旧的ITL文件，而新密钥不在电话上的ITL文件中。
2. 由于电话在旧ITL中找不到新的CCM+TFTP签名，因此它尝试联系TVS服务。

> 注意：这部分极其重要。旧ITL文件的TVS证书必须仍然匹配。如果同时重新生成CallManager.pem和TVS.pem，则电话无法下载任何新文件，除非从电话上手动删除ITL。

3. 当电话与TVS联系时，运行TVS的CUCM服务器在操作系统证书存储区中具有新的CallManager.pem证书。
4. TVS服务器返回成功，电话将新的ITL文件加载到内存中。
5. 电话现在尝试下载已使用新的CallManager.pem密钥签名的配置文件。
6. 自加载新的ITL以来，新签名的配置文件由内存中的ITL成功验证。

要点:

- 切勿同时重新生成CallManager.pem和TVS.pem证书。
- 如果重新生成TVS.pem或CallManager.pem，则必须重新启动TVS和TFTP并重置电话，才能获取新的ITL文件。
- CUCM的较新版本会自动处理此电话重置，并在证书重新生成时警告用户。
- 如果存在多个TVS服务器（CallManager组中存在多个服务器），则其他服务器可以对新的CallManager.pem证书进行身份验证。

在集群之间移动电话

当您将ITL就绪的电话从一个集群移动到另一个集群时，必须考虑ITL和TFTP私钥。

提供给电话的任何新配置文件都必须与CTL、ITL中的签名或电话当前TVS服务中的签名匹配。

本文档说明如何确保新的集群ITL文件和配置文件可受电话上当前ITL文件的信任。
https://supportforums.cisco.com/docs/DOC-15799。

备份和恢复

CallManager.pem证书和私钥通过灾难恢复系统(DRS)进行备份。 如果重建TFTP服务器，则必须从备份中将其恢复，以便可以恢复私钥。

如果服务器上没有CallManager.pem私钥，使用旧密钥的当前ITL电话不会信任已签名的配置文件。

如果重建了群集并且未从备份中恢复，则它与“在群集之间移动电话”文档完全相同。这是因为对于

电话而言，包含新密钥的集群是不同的集群。

有一个与备份和恢复相关的严重缺陷。如果群集易受Cisco Bug ID CSCtn50405，则DRS备份不包含CallManager.pem证书。

这会导致从此备份还原的任何服务器生成损坏的ITL文件，直到生成新的CallManager.pem。

如果没有其他正常运行的TFTP服务器未完成备份和还原操作，这可能意味着需要从电话中删除所有ITL文件。

要验证是否需要重新生成CallManager.pem文件，请输入show itl命令，然后输入：

```
run sql select c.subjectname, c.serialnumber, c.ipv4address, t.name from
certificate as c, certificatetrustrolemap as r, typetrustrole as t where  c.pkid =
r.fkcertificate and t.enum = r.tktrustrole
```

在ITL输出中，要查找的主要错误包括：

```
This etoken was not used to sign the ITL file.
```

和

```
Verification of the ITL file failed.
Error parsing the ITL file!!
```

以前的结构化查询语言(SQL)查询搜索具有“身份验证和授权”角色的证书。

具有身份验证和授权角色的上一个数据库查询中的CallManager.pem证书也必须出现在操作系统管理证书管理网页中。

如果遇到之前的缺陷，则查询和操作系统网页中的CallManager.pem证书不匹配。

更改主机名或域名

如果更改CUCM服务器的主机名或域名，它会立即在该服务器上重新生成所有证书。证书重新生成部分解释说，重新生成TVS.pem和CallManager.pem是一件“坏事”。

主机名更改失败的情况有几种，而主机名更改正常运行的情况也有几种。本部分介绍所有这些功能，并将它们链接回您已从本文档了解的有关TVS和ITL的内容。

仅使用ITL的单节点集群（请注意，此操作会中断，不会进行准备）

- 使用Business Edition服务器或仅发布者部署，在您更改主机名的同时重新生成CallManager.pem和TVS.pem。
- 如果在单个节点集群上更改主机名，而不首先使用此处介绍的Rollback Enterprise参数，则电话无法根据当前ITL文件验证新的ITL文件或配置文件。
- 电话无法连接到TVS，因为TVS证书也不再受信任。
- 电话显示有关“信任列表验证失败”的错误，新的配置更改不会生效，安全服务URL将失败。
- 如果不首先采取步骤2中的预防措施，唯一的解决办法是从每部电话上手动删除国际交易日志。

同时具有CTL和ITL的单节点集群（可以暂时中断，但容易修复）

- 执行服务器重命名操作后，重新运行CTL客户端。这会将新的CallManager.pem证书放入电话下载的CTL文件中。
- 根据CTL文件中的CCM+TFTP功能，可以信任包含新ITL文件的新配置文件。
- 由于更新的CTL文件基于保持不变的USB eToken私钥受信任，因此此方法有效。

仅使用ITL的多节点集群（这通常有效，但如果匆忙执行，可能会永久断开）

- 由于多节点集群有多台TVS服务器，因此任何一台服务器都可以毫无问题地重新生成其证书。当电话收到此新的不熟悉的签名时，它会要求其他TVS服务器验证新的服务器证书。
- 导致此过程失败的主要原因有两个：
  - 如果同时重命名并重新启动所有服务器，则当服务器和电话恢复正常时，无法通过已知证书访问任何TVS服务器。
  - 如果电话在CallManager组中只有一个服务器，则其他TVS服务器不会造成任何差异。请参阅“单节点集群”场景以解决此问题，或将其他服务器添加到电话CallManager组。

同时具有CTL和ITL的多节点集群（无法永久断开）

- 在运行重命名后，TVS服务会对新证书进行身份验证。
- 即使所有TVS服务器因某种原因不可用，仍可使用CTL客户端更新具有新CallManager.pem CCM+TFTP证书的电话。

集中式TFTP

当带有ITL的电话启动时，它会请求以下文件：CTLSEP<MAC Address>.tlv、ITLSEP<MAC Address>.tlv和SEP<MAC Address>.cnf.xml.sgn。

如果电话找不到这些文件，则它会请求ITLFile.tlv和CTLFile.tlv,CTLFile.tlv是集中式TFTP服务器提供给任何请求它的电话的。

使用集中式TFTP时，有一个TFTP集群指向多个其他子集群。

通常，这是因为多个CUCM集群上的电话共享相同的DHCP作用域，因此必须具有相同的DHCP选项150 TFTP服务器。

所有IP电话都指向中央TFTP集群，即使它们注册到其他集群也是如此。此中央TFTP服务器在收到其无法找到文件的请求时查询远程TFTP服务器。

由于这种操作，集中式TFTP只能在ITL同构环境中运行。

所有服务器必须运行CUCM 8.x版或更高版本，或者所有服务器必须运行8.x版之前的版本。

如果集中TFTP服务器提供ITLFile.tlv，则电话不信任来自远程TFTP服务器的任何文件，因为签名不匹配。

这发生在不同的混合体中。在同类混合中，电话请求ITLSEP<MAC>.tlv，该请求是从正确的远程集群中提取的。

在混合使用早于8.x版和早于8.x版的集群的异类环境中，必须在8.x版集群上启用“准备集群以回滚到8.0版”，如Cisco bug ID CSCto87262中所述。

使用HTTP而不是HTTPS配置“安全电话URL参数”。这将有效禁用电话上的ITL功能。

常见问题解答

我可以关闭SBD吗？

您只能在SBD和ITL当前工作的情况下关闭SBD。

使用Prepare Cluster for Rollback to pre 8.0" Enterprise Parameter和使用HTTP而不是HTTPS配置“Secured Phone URL Parameters”可在电话上临时禁用SBD。

当您设置Rollback参数时，它会创建一个带有空函数条目的签名ITL文件。

“空”ITL文件仍被签名，因此集群必须处于完全正常运行的安全状态，才能启用此参数。

启用此参数并下载并验证带有空白条目的新ITL文件后，电话接受任何配置文件，无论其签署人是谁。

建议不要将群集保持此状态，因为之前提到的三个功能（经过身份验证的配置文件、加密的配置文件和HTTPS URL）均不可用。

CallManager.pem丢失后，能否从所有电话轻松删除ITL文件？

目前没有从思科远程提供的电话中删除所有ITL的方法。正因如此，本文档中介绍的程序和交互必须予以考虑。

Cisco Bug ID CSCto47052当前有一个尚未解决的增强功能，它请求此功能，但尚未实施。

在过渡期间，已通过Cisco Bug ID CSCts01319添加了一项新功能，该功能可能允许思科技术支持中心(TAC)恢复到之前受信任的ITL（如果服务器中仍可用）。

这仅适用于集群位于具有此缺陷修复程序的版本中的某些情况，以及之前的ITL存在于存储在服务器上的特定位置的备份中。

查看缺陷以查看您的版本是否有修补程序。请与Cisco TAC联系，以便完成缺陷中说明的可能的恢复过程。

如果不能执行上述程序，则必须手动按下电话上的电话按钮，以便删除ITL文件。这是安全与易于管

理之间的权衡。要保证国际交易日志文件的真正安全，就必须不容易将其远程删除。

即使使用脚本按钮按下简单对象访问协议(SOAP)XML对象，也无法远程删除ITL。

这是因为，此时，TVS访问（因此用于验证传入SOAP XML按钮按钮按钮对象的安全身份验证URL访问）不起作用。

如果身份验证URL未配置为安全，可以编写用于删除ITL的按键脚本，但思科不提供此脚本。

其他不使用身份验证URL对远程按键编写脚本的方法可能由第三方提供，但这些应用程序不由思科提供。

删除国际交易日志的最常用方法是向所有电话用户发送电子邮件广播，指示他们输入按键序列。

如果设置访问权限设置为Restricted或Disabled，则电话需要重置为出厂设置，因为用户无权访问电话的“设置”菜单。

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）。

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）。