在安全群集之间迁移电话

目录

简介

本文档介绍如何在两个安全的Cisco Unified Communications Manager(CUCM)群集之间迁移电话。

作者：思科TAC工程师David Norman。

先决条件

要求

思科建议您了解CUCM。

使用的组件

本文档中的信息基于以下软件版本：

源群集：CUCM 版本 10.5.2.11900-3
目标群集： CUCM 版本 11.0.1.10000-10
使用固件sip88xx.10-3-1-20的8861电话

证书信任列表(CTL)文件使用CallManager证书（而非USB令牌）签名

背景

在迁移过程中，电话尝试建立与源群集思科信任验证服务(TVS)的安全连接，以验证目标群集CallManager证书。如果电话的证书信任列表(CTL)和身份信任列表(ITL)文件无效，则电话无法与TVS完成安全握手，迁移到目标群集将不会成功。在开始电话迁移过程之前，请确认电话安装了正确的CTL/ITL文件。此外，在源群集上，确认企业功能“准备群集以回滚到8.0之前版本”设置为False。

配置

将目标群集CallManager证书导入源群集CallManager-trust和Phone-SAST-trust存储。有两种方法可以做到这一点。

方法 1.

使用批量证书工具并在源群集和目标群集上完成以下步骤。

步骤1.在源和目标集**群上导航**到Cisco Unified OS Administration页 **> Security > Bulk Certificate Management**。

步骤2.输入安全文件传输协议(SFTP)服务器的详细信息，然后选择**保存**。

步骤3.选择**Export**并导出简单文件传输协议(TFTP)证书。

步骤4.单击“合并”**按钮**以执行证书合并。 这将创建一个PKCS12文件，其中包括源和目标CallManager证书。

步骤5.将整合的证书导入到每个集群。

在整合过程中（步骤5），源群集CallManager证书上传到CallManager-trust和Phone-SAST-trust存储中的目标群集。这允许电话迁移回源群集。如果遵循手动方法，源群集CallManager证书 不会 上传到目标集群。这意味着您无法将电话迁移回源群集。如果希望选择将电话迁移回源群集，则 需要将源群集CallManager证书上传到目标群集CallManager-trust和Phone-SAST-trust存储。

> **注意：**两个集群必须将TFTP证书导出到同一SFTP服务器和同一SFTP目录。

> **注意：**仅一个集群上需要步骤4。如果在CUCM 8.x或9.x版之间将电话迁移到CUCM 10.5.2.13900-12或更高版本，请在整合证书之前记下此Cisco Bug ID CSCuy43181。

方法 2.

手动导入证书。在目标集群上完成以下步骤。

步骤1.导航至**Cisco Unified OS Administration页**> Security > Certificate Management。

步骤2.选择CallManager.pem证书并下载。

步骤3.选择ITLrecovery.pem证书并下载

步骤4.将CallManager证书作为CallManger-trust和Phone-SAST-trust证书上传到源群集发布者。

步骤5.将ITLrecovery证书作为Phone-SAST-Trust上传到源群集

步骤6.从源群集重新启动所有节点中的TVS。

然后，证书将复制到群集中的其他节点。

第3、5、6步适用于将电话从8.x迁移到12.x的场景

> **注意：**需要从目标群集上运行TFTP服务的所有节点下载CallManager证书。

使用上述方法之一上传证书后，请更改电话动态主机配置协议(DHCP)选项150以指向目标集群的TFTP地址。

> **警告：**在非安全群集之间迁移电话的一种方法是在源群集上将“准备群集以回滚到8.0之前”设置为True并重新启动电话。当您在安全群集之间迁移电话时，不能选择此选项。这是因为回滚到8.0之前版本的功能只会空出ITL文件（它不会空出CTL文件）。 这意味着当电话迁移并从目标群集下载CTL文件时，它需要使用源群集TVS验证新的CTL。由于电话的ITL文件不包含源群集TVS证书，因此当电话尝试建立与TVS服务的安全连接时握手失败。

验证

这是来自源群集的电话控制台日志和TVS日志（设置为详细）的摘录。该片段显示了电话注册到目标群集的过程。

1.电话从目标群集启动并下载CTL文件。

```
3232 NOT Nov 29 06:33:59.011270 downd-DDFORK - execing [/usr/sbin/dgetfile][-L620][ ]
3233 NOT Nov 29 06:33:59.033132 dgetfile(870)-GETXXTP
[GT870][src=CTLSEPB000B4BA0AEE.tlv][dest=/tmp/CTLFile.tlv][serv=][serv6=][sec=0]
```

2. CTL文件由目标集群呼叫管理器证书签名，该证书不在电话现有CTL或ITL文件中。这意味着电话需要联系其TVS服务以验证证书。此时，电话仍具有其旧配置，其中包含源集群TVS服务的IP地址（电话配置中指定的TVS与电话呼叫管理器组相同）。 电话建立到TVS服务的SSL连接。当TVS服务向电话提供其证书时，电话会根据其ITL文件中的证书验证证书。如果它们相同，则握手成功完成。

```
3287 INF Nov 29 06:33:59.395199 SECUREAPP-Attempting connect to TVS server addr [192.168.11.32],
mode [IPv4]
3288 INF Nov 29 06:33:59.395294 SECUREAPP-TOS set to [96] on sock, [192.168.11.32][11]
3289 INF Nov 29 06:33:59.396011 SECUREAPP-TCP connect() successful, [192.168.11.32] [11]
3290 DEB Nov 29 06:33:59.396111 SECUREAPP-BIO created with: addr:192.168.11.32, port:2445,
mode:IPv4
3291 INF Nov 29 06:33:59.396231 SECUREAPP-Sec SSL Connection - TVS.
3292 INF Nov 29 06:33:59.396379 SECUREAPP-SSL session setup - Requesting Cert
3293 DEB Nov 29 06:33:59.396402 SECUREAPP-Obtaining certificate.
3294 INF Nov 29 06:33:59.396444 SECUREAPP-SSL session setup - Get Active cert ok
3295 DEB Nov 29 06:33:59.396464 SECUREAPP-SSL session setup - cert len=785, type=LSC
3296 DEB Nov 29 06:33:59.396854 SECUREAPP-Certificate subject name = /serialNumber=PID:CP-8861
SN:FCH18198CNQ/C=AU/O=stormin/OU=IST/CN=CP-8861-SEPB000B4BA0AEE
3297 DEB Nov 29 06:33:59.396917 SECUREAPP-SSL session setup - Certificate issuer name =
/C=AU/O=stormin/OU=IST/CN=CAPF-a7fb32bf/ST=NSQ/L=Sydney
3298 INF Nov 29 06:33:59.396947 SECUREAPP-SSL session setup - Requesting Pkey
3299 INF Nov 29 06:33:59.397024 SECUREAPP-SSL session setup - Get private key ok
3300 DEB Nov 29 06:33:59.397045 SECUREAPP-SSL session setup - key len=1191
3301 INF Nov 29 06:33:59.399181 SECUREAPP-Setup SSL session - SSL use certificate okay
3302 INF Nov 29 06:33:59.399477 SECUREAPP-Setup SSL session - SSL use private key okay
3303 DEB Nov 29 06:33:59.399974 SECUREAPP-Sec SSL Connection - Added SSL connection handle
0x40e01270, connDesc 11 to table.
3304 DEB Nov 29 06:33:59.400225 SECUREAPP-Sec SSL Connection - check status & perform handshake.
3305 DEB Nov 29 06:33:59.401086 SECUREAPP-Blocked TVS Secure Connection - Waiting (0) ....
3306 DEB Nov 29 06:33:59.401796 SECUREAPP-Sec SSL Connection - check status & perform handshake.
3307 DEB Nov 29 06:33:59.403321 SECUREAPP-SSL session setup Cert Verification - Role is = 21
3308 INF Nov 29 06:33:59.403412 SECUREAPP-SSL session setup Cert Verification - Invoking
```

```
certificate validation helper plugin.
3309 INF Nov 29 06:33:59.403662 SECUREAPP-SSL session setup Cert Verification - Certificate
validation helper plugin returned.
3310 INF Nov 29 06:33:59.403731 SECUREAPP-SSL session setup Cert Verification - Certificate is
valid.
3311 DEB Nov 29 06:33:59.403784 SECUREAPP-SSL session setup Cert Verification - returning
validation result = 1
3312 ERR Nov 29 06:33:59.428892 downd-SOCKET accept errno=4 "Interrupted system call"
3313 DEB Nov 29 06:33:59.907337 SECUREAPP-Blocked TVS Secure Connection - Waiting (1) ....
3314 DEB Nov 29 06:33:59.907393 SECUREAPP-Sec SSL Connection - check status & perform handshake.
3315 NOT Nov 29 06:33:59.908586 SECUREAPP-Sec SSL Connection - Handshake successful.
3316 INF Nov 29 06:33:59.908696 SECUREAPP-Sec SSL Connection - caching disabled, session not
saved
3317 DEB Nov 29 06:33:59.908752 SECUREAPP-Connection to server succeeded
```

3. TVS日志显示来自电话的传入连接，握手成功。

```
18:01:05.333 | debug Accepted TCP connection from socket 0x00000012, fd = 8
18:01:05.333 | debug Total Session attempted = 7 accepted = 7
18:01:05.334 | debug tvsGetNextThread
18:01:05.334 | debug Recd event
18:01:05.334 | debug new ph on fd 8
18:01:05.334 | debug 7:UNKNOWN:Got a new SCB from RBTree
18:01:05.334 | debug ipAddrStr (Phone) 192.168.11.100
18:01:05.334 | debug 8:UNKNOWN:Got a new ph conn 192.168.11.100 on 8, Total Acc = 7..
18:01:05.334 | debug addded 8 to readset
18:01:05.338 | debug after select, 8 was set
18:01:05.338 | debug ipAddrStr (Phone) 192.168.11.100
18:01:05.855 | debug tvsSSLHandShakeNotify
18:01:05.855 | debug 192.168.11.100: tvsSSLHandShake Session ciphers - AES256-SHA
18:01:05.855 | debug addded 8 to readset
18:01:05.855 | debug Recd event
18:01:05.855 | debug TLS HS Done for ph_conn
```

4.电话控制台日志显示电话向TVS服务发送请求，以从目标集群验证呼叫管理器证书。

```
3318 DEB Nov 29 06:33:59.908800 SECUREAPP-TVS provider Init - connect returned TVS srvr sock: 11
3319 DEB Nov 29 06:33:59.908848 SECUREAPP-TVS process request - processing TVS Query Certificate
request.
3320 NOT Nov 29 06:33:59.909322 SECUREAPP-TVS process request - Successfully sent the TVS
request to TVS server, bytes written : 153
3321 DEB Nov 29 06:33:59.909364 SECUREAPP-==== TVS process request - request byte dump ====, len
= 153
3322 DEB Nov 29 06:33:59.913075 SECUREAPP-TVS Service receives 1480 bytes of data
3323 DEB Nov 29 06:33:59.913270 SECUREAPP-==== TVS process response - response byte dump ====,
len = 1480
3324 DEB Nov 29 06:33:59.914466 SECUREAPP-Found the work order from pending req list element at
index 0
```

5. TVS日志显示收到请求。

```
18:01:06.345 | debug 8:UNKNOWN:Incoming Phone Msg:
HEX_DUMP: Len = 153:
```

```
18:01:06.345 | debug 57 01 03 00 00 00 03 e9
18:01:06.345 | debug 00 8f 01 00 18 01 43 50
18:01:06.345 | debug 2d 38 38 36 31 2d 53 45
18:01:06.345 | debug 50 42 30 30 30 42 34 42
18:01:06.345 | debug 41 30 41 45 45 03 00 42
18:01:06.345 | debug 43 4e 3d 75 63 6d 31 31
18:01:06.345 | debug 70 75
18:01:06.345 | debug tvsPhoneDecodeMsg -
Decoded Phone Msg:
18:01:06.345 | debug Protocol Discriminator: 57
18:01:06.345 | debug MsgType : TVS_MSG_QUERY_CERT_REQ
18:01:06.345 | debug Session Id : 0
18:01:06.345 | debug Length : 143
18:01:06.345 | debug 8:UNKNOWN:TVS CORE: Rcvd Event: TVS_EV_QUERY_CERT_REQ in State:
TVS_STATE_AWAIT_REQ
18:01:06.345 | debug tvsHandleQueryCertReq
18:01:06.345 | debug tvsHandleQueryCertReq : Subject Name is:
CN=ucm11pub.stormin.local;OU=IST;O=Stormin;L=Brisbane;ST=QLD;C=AU
18:01:06.345 | debug tvsHandleQueryCertReq : Issuer Name is: CN=stormin-WIN2012-CA
18:01:06.345 | debug tvsHandleQueryCertReq : Serial Number is:
24000000179479B8F124AC3F3B000000000017
18:01:06.345 | debug CertificateDBCache::getCertificateInformation - Looking up the certificate
cache using Unique MAP ID : 24000000179479B8F124AC3F3B000000000017CN=stormin-WIN2012-CA
18:01:06.345 | debug CertificateDBCache::getCertificateInformation - Found entry {rolecount : 2}
18:01:06.345 | debug CertificateDBCache::getCertificateInformation - {role : 0}
18:01:06.346 | debug CertificateDBCache::getCertificateInformation - {role : 3}
18:01:06.346 | debug convertX509ToDER -x509cert : 0xbb696e0
```

6. TVS日志显示其存储中的证书，TVS向电话发送响应。

```
18:01:06.346 | debug 8:UNKNOWN:Sending QUERY_CERT_RES msg
18:01:06.346 | debug tvsPhoneDecodeMsg -
Decoded Phone Msg:
18:01:06.346 | debug Protocol Discriminator: 57
18:01:06.346 | debug MsgType : TVS_MSG_QUERY_CERT_RES
18:01:06.346 | debug Session Id : 0
18:01:06.346 | debug Length : 1470
18:01:06.346 | debug ReasonInfo : 00$
18:01:06.346 | debug Number of Certs : 1
18:01:06.346 | debug Cert[0] :
18:01:06.346 | debug Cert Type : 0
HEX_DUMP: Len = 1451:
18:01:06.346 | debug 30 82 05 a7 30 82 04 8f
18:01:06.346 | debug a0 03 02 01 02 02 13 24
18:01:06.346 | debug 00 00 00 17 94 79 b8 f1
18:01:06.346 | debug 24 ac 3f 3b 00 00 00 00
18:01:06.346 | debug 00 17 30 0d 06 09 2a 86
18:01:06.346 | debug 48 86 f7 0d 01 01 0b 05
18:01:06.346 | debug 00 30
18:01:06.346 | debug Version : 0
18:01:06.346 | debug PublicKey :
HEX_DUMP: Len = 4:
18:01:06.347 | debug 00 01 51 80
18:01:06.347 | debug Sending TLS Msg ..
HEX_DUMP: Len = 1480:
18:01:06.347 | debug 57 01 04 f7 00 00 03 e9
18:01:06.347 | debug 05 be 07 00 01 00 02 05
18:01:06.347 | debug ab 30 82 05 a7 30 82 04
18:01:06.347 | debug 8f a0 03 02 01 02 02 13
```

```
18:01:06.347 | debug 24 00 00 00 17 94 79 b8
18:01:06.347 | debug f1 24 ac 3f 3b 00 00 00
18:01:06.347 | debug 00 00
18:01:06.347 | debug ipAddrStr (Phone) 192.168.11.100
```

7.电话控制台日志显示证书已成功验证，CTL文件已更新。

```
3325 INF Nov 29 06:33:59.915121 SECUREAPP-TVS added cert to TVS cache - expires in 24 hours
3333 NOT Nov 29 06:34:00.411671 SECUREAPP-Hashes match... authentication successful.
3334 WRN Nov 29 06:34:00.412849 SECUREAPP-AUTH: early exit from parser loop; old version header?
3335 WRN Nov 29 06:34:00.412945 SECUREAPP-AUTH: hdr ver 1.2 (knows only upto 1.1)
3336 NOT Nov 29 06:34:00.413031 SECUREAPP-updateFromFile: TL parse to table: CTL_SUCCESS
3337 NOT Nov 29 06:34:00.413088 SECUREAPP-updateFromFile: Updating master TL table
3338 DEB Nov 29 06:34:00.413442 SECUREAPP-TL file verified successfully.
3339 INF Nov 29 06:34:00.413512 SECUREAPP-TL file updated.
```

8.电话控制台日志显示电话下载其ITL文件的时间。

```
3344 NOT Nov 29 06:34:00.458890 dgetfile(877)-GETXXTP
[GT877][src=ITLSEPB000B4BA0AEE.tlv][dest=/tmp/ITLFile.tlv][serv=][serv6=][sec=0]
3345 NOT Nov 29 06:34:00.459122 dgetfile(877)-In normal mode, call - > makeXXTPrequest (V6...)

3281 NOT Dec 14 06:34:00.488697 dgetfile(851)-XXTP complete - status = 100
3282 NOT Dec 14 06:34:00.488984 dgetfile(851)-XXTP actualserver [192.168.11.51]
```

9.根据CTL文件验证ITL文件。CTL文件包含目标群集CallManager证书。这意味着电话可以验证证书，而无需联系源群集TVS服务。

```
3287 NOT Nov 29  06:34:00.499372 SECUREAPP-Hashes match... authentication successful.
3288 WRN Nov 29  06:34:00.500821 SECUREAPP-AUTH: early exit from parser loop; old version
header?
3289 WRN Nov 29 06:34:00.500987 SECUREAPP-AUTH: hdr ver 1.2 (knows only upto 1.1)
3290 NOT Nov 29  06:34:00.501083 SECUREAPP-updateFromFile: TL parse to table: CTL_SUCCESS
3291 NOT Nov 29  06:34:00.501147 SECUREAPP-updateFromFile: Updating master TL table
3292 DEB Nov 29  06:34:00.501584 SECUREAPP-TL file verified successfully.
3293 INF Nov 29  06:34:00.501699 SECUREAPP-TL file updated.
```

故障排除

在迁移过程之前，验证电话上的CTL/ITL。有关如何验证CTL/ITL的详细信息，请访问：https://www.cisco.com/c/en/us/support/docs/voice-unified-communications/unified-communications-manager-callmanager/116232-technote-sbd-00.html#anc9