Migre telefones entre clusters seguros

Contents

[Introduction](#)
[Prerequisites](#)
[Requirements](#)
[Componentes Utilizados](#)
[Background](#)
[Configurar](#)
[Verificar](#)
[Troubleshoot](#)

Introduction

Este documento descreve como migrar telefones entre dois clusters seguros do Cisco Unified Communications Manager (CUCM).

Contribuído por David Norman, engenheiro do Cisco TAC.

Prerequisites

Requirements

A Cisco recomenda que você tenha conhecimento do CUCM.

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software:

Cluster de origem: CUCM versão 10.5.2.11900-3
Cluster de destino: CUCM versão 11.0.1.10000-10
Telefone 8861 usando o firmware sip88xx.10-3-1-20

Os arquivos CertificateTrust List (CTL) são assinados com o certificado do CallManager (e não com o token USB)

Background

Durante o processo de migração, o telefone tenta configurar uma conexão segura com os clusters de origem do Cisco Trust Verification Service (TVS) para verificar o certificado CallManager dos clusters de destino. Se o CTL (Certificate Trust List) e o arquivo Identity Trust List (ITL) do telefone forem inválidos, o telefone não poderá concluir o handshake seguro com o TVS e a migração para o cluster de destino não será bem-sucedida. Antes de iniciar o processo de migração do telefone, confirme se os telefones têm o arquivo CTL/ITL correto instalado. Também no cluster de origem, confirme se o recurso empresarial "Prepare o cluster para reversão para Pre

8.0" está definido como Falso.

Configurar

Importar o certificado do CallManager dos clusters de destino para o armazenamento de confiança CallManager-trust e Phone-SAST. Há dois métodos para fazer isso.

Método 1.

Use a Bulk Certificate Tool e conclua estas etapas nos clusters de origem e de destino.

Etapa 1. Navegue até a página **Cisco Unified OS Administration > Security > Bulk Certificate Management** em clusters de origem e de destino.

Etapa 2. Insira os detalhes do servidor Secure File Transfer Protocol (SFTP) e selecione **Save**.

Etapa 3. Selecione **Exportar** e exporte o certificado TFTP (Trivial File Transfer Protocol).

Etapa 4. Clique no botão **Consolidar** para executar a consolidação do certificado. Isso cria um arquivo PKCS12 que inclui o certificado origem e o certificado destino do CallManager.

Etapa 5. Importar os certificados consolidados de volta para cada cluster.

Durante o processo de consolidação (Etapa 5), o clusters de origem O certificado do CallManager é carregado para o cluster de destino no armazenamento de confiança do CallManager e do Phone-SAST-trust. Isso permite que os telefones migrem de volta para o cluster de origem. Se o método manual for seguido, a origem agrupa o certificado do CallManager não vai ser carregado no cluster de destino. Isso significa que você não pode migrar os telefones de volta para o cluster de origem. Se desejar que a opção migre os telefones de volta para o cluster de origem, você é necessário carregar o certificado do CallManager dos clusters de origem para o armazenamento de confiança CallManager-trust e Phone-SAST-trust dos clusters de destino.

> **Note**: Ambos os clusters devem exportar o certificado TFTP para o mesmo servidor SFTP e para o mesmo diretório SFTP.

> **Note**: A etapa 4 é necessária somente em um cluster. Se você migrar telefones entre CUCM versão 8.x ou 9.x para CUCM versão 10.5.2.13900-12 ou mais recente, anote este ID de bug da Cisco CSCuy43181 antes de consolidar os certificados.

Método 2.

Importar manualmente os certificados. Conclua estas etapas no cluster de destino.

Etapa 1. Navegue até a página **Cisco Unified OS Administration > Security > Certificate Management**.

Etapa 2. Selecione o certificado CallManager.pem e baixe-o.

Etapa 3. Selecione o certificado ITLrecovery.pem e baixe-o

Etapa 4. Carregue o certificado do CallManager para o editor do cluster de origem como um certificado CallManager-trust e Phone-SAST-trust.

Etapa 5. Carregue o certificado de recuperação ITL para o cluster de origem como Phone-SAST-Trust

Etapa 6. Reinicie o TVS em todos os nós do cluster de origem.

Em seguida, os certificados são replicados para os outros nós no cluster.

As etapas 3, 5 e 6 serão aplicadas a cenários de migração de telefone de 8.x para 12.x

> **Note**: O certificado do CallManager precisa ser baixado de todos os nós que executam o serviço TFTP no cluster de destino.

Depois que os certificados tiverem sido carregados com um dos métodos acima, altere a Opção 150 do Dynamic Host Configuration Protocol (DHCP) para apontar para o endereço TFTP dos clusters de destino.

> **Caution**: Um método para migrar telefones entre clusters não seguros é definir "Prepare Cluster para Reversão para pré 8.0" como True no cluster de origem e reinicie os telefones. Esta não é uma opção quando você migra telefones entre clusters seguros. Isso ocorre porque a reversão para o recurso anterior à 8.0 só libera o arquivo ITL (não deixa em branco o arquivo CTL). Isso significa que quando o telefone é migrado e faz o download do arquivo CTL do cluster de destino, ele precisa verificar a nova CTL com os clusters de origem TVS. Como o arquivo ITL do telefone não contém o certificado TVS do cluster de origem, o handshake falha quando o telefone tenta estabelecer uma conexão segura com o serviço TVS.

Verificar

Este é um trecho dos registros do console do telefone e dos registros TVS (definidos como detalhados) do cluster de origem. Os trechos mostram o processo de registro dos telefones no cluster de destino.

1. O telefone inicializa e baixa o arquivo CTL do cluster de destino.

```
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. O arquivo CTL é assinado pelo certificado do gerenciador de chamadas dos clusters de destino que não está no arquivo CTL ou ITL existente dos telefones. Isso significa que o telefone precisa entrar em contato com o serviço TVS para verificar o certificado. Neste ponto, o telefone ainda tem sua configuração antiga que contém o endereço IP do serviço TVS do cluster de origem (o TVS especificado na configuração dos telefones é o mesmo que o grupo do gerenciador de chamadas dos telefones). O telefone configura uma conexão SSL para o serviço TVS. Quando o serviço TVS apresenta seu certificado ao telefone, o telefone verifica o certificado em relação ao

certificado em seu arquivo ITL. Se forem iguais, o handshake é concluído com êxito.

```
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. Os registros TVS mostram a conexão de entrada do telefone e o handshake foi bem-sucedido.

```
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. Os registros do console do telefone mostram que o telefone envia uma solicitação ao serviço TVS para verificar o certificado do gerenciador de chamadas do cluster de destino.

```
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. Os registros TVS mostram que a solicitação foi recebida.

```
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. Os registros TVS mostram o certificado em sua loja e o TVS envia uma resposta ao telefone.

```
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. Os registros do console do telefone mostram que o certificado foi verificado com êxito e que o arquivo CTL foi atualizado.

```
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. Os registros do console do telefone são mostrados quando o telefone baixa seu arquivo ITL.

```
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. O arquivo ITL é verificado em relação ao arquivo CTL. O arquivo CTL contém o certificado do CallManager dos clusters de destino. Isso significa que o telefone pode verificar o certificado sem entrar em contato com o serviço TVS do cluster de origem.

```
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.
```

Troubleshoot

Antes do processo de migração, verifique a CTL/ITL nos telefones. Mais informações sobre como verificar o CTL/ITL podem ser encontradas
aqui: https://www.cisco.com/c/en/us/support/docs/voice-unified-communications/unified-communications-manager-callmanager/116232-technote-sbd-00.html#anc9