Segurança do CUCM por padrão e operação e solução de problemas do ITL

Contents

Introduction

Este documento descreve o recurso Segurança por padrão (SBD) do Cisco Unified Communications Manager (CUCM) versões 8.0 e posteriores. Este documento serve como um complemento aos documentos oficiais Security By Default, e fornece informações operacionais e dicas de solução de problemas para ajudar os administradores e facilitar o processo de solução de problemas.

Informações de Apoio

O CUCM Versão 8.0 e posterior apresenta o recurso SBD, que consiste em arquivos de Lista de Confiança de Identidade (ITL - Identity Trust List) e o Serviço de Verificação de Confiança (TVS -

Trust Verification Service). Cada cluster do CUCM agora usa a segurança baseada em ITL automaticamente. Há uma troca entre segurança e facilidade de uso/administração que os administradores devem conhecer antes de fazer determinadas alterações em um cluster CUCM versão 8.0.

Éuma boa ideia familiarizar-se com esses conceitos básicos de SBD: Artigo da Wikipedia Asymmetric Key Cryptography e artigo da Wikipedia Public Key Infrastructure.

Visão geral do SBD

Esta seção fornece uma rápida visão geral do que exatamente o SBD oferece. Para obter detalhes técnicos completos de cada função, consulte a seção Detalhes do SBD e Informações sobre Troubleshooting.

O SBD fornece estas três funções para telefones IP suportados:

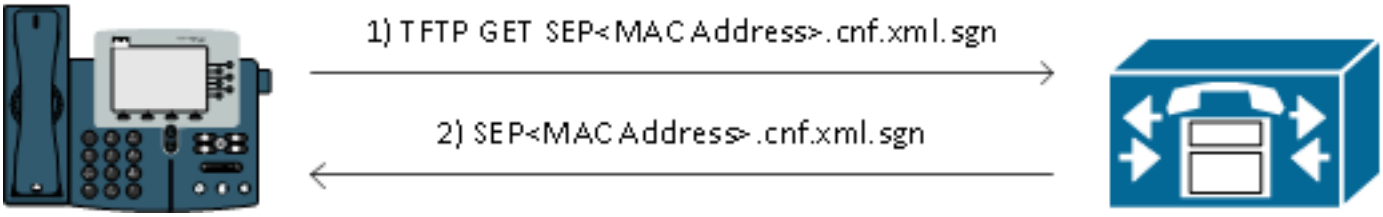
- Autenticação padrão dos arquivos de download do TFTP (configuração, localização, lista de ingressos) que usam uma chave de assinatura
- Criptografia opcional de arquivos de configuração TFTP que usam uma chave de assinatura
- Verificação de certificado para conexões HTTPS iniciadas pelo telefone que usam um repositório confiável de certificado remoto no CUCM (TVS)

Este documento fornece uma visão geral de cada uma dessas funções.

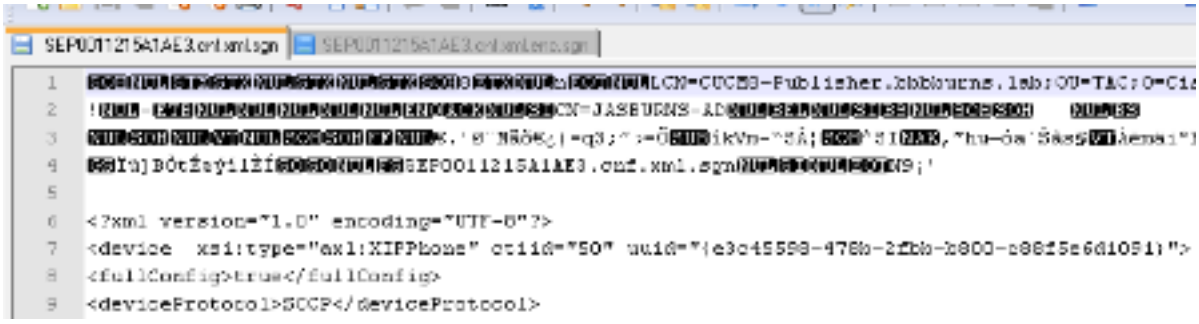
Autenticação de download de TFTP

Quando um arquivo CTL (Certificate Trust List) ou ITL está presente, o telefone IP solicita um arquivo de configuração TFTP assinado do servidor TFTP CUCM. Esse arquivo permite que o telefone verifique se o arquivo de configuração veio de uma fonte confiável. Com os arquivos CTL/ITL presentes nos telefones, os arquivos de configuração devem ser assinados por um servidor TFTP confiável. O arquivo é texto simples na rede enquanto é transmitido, mas vem com uma assinatura de verificação especial.

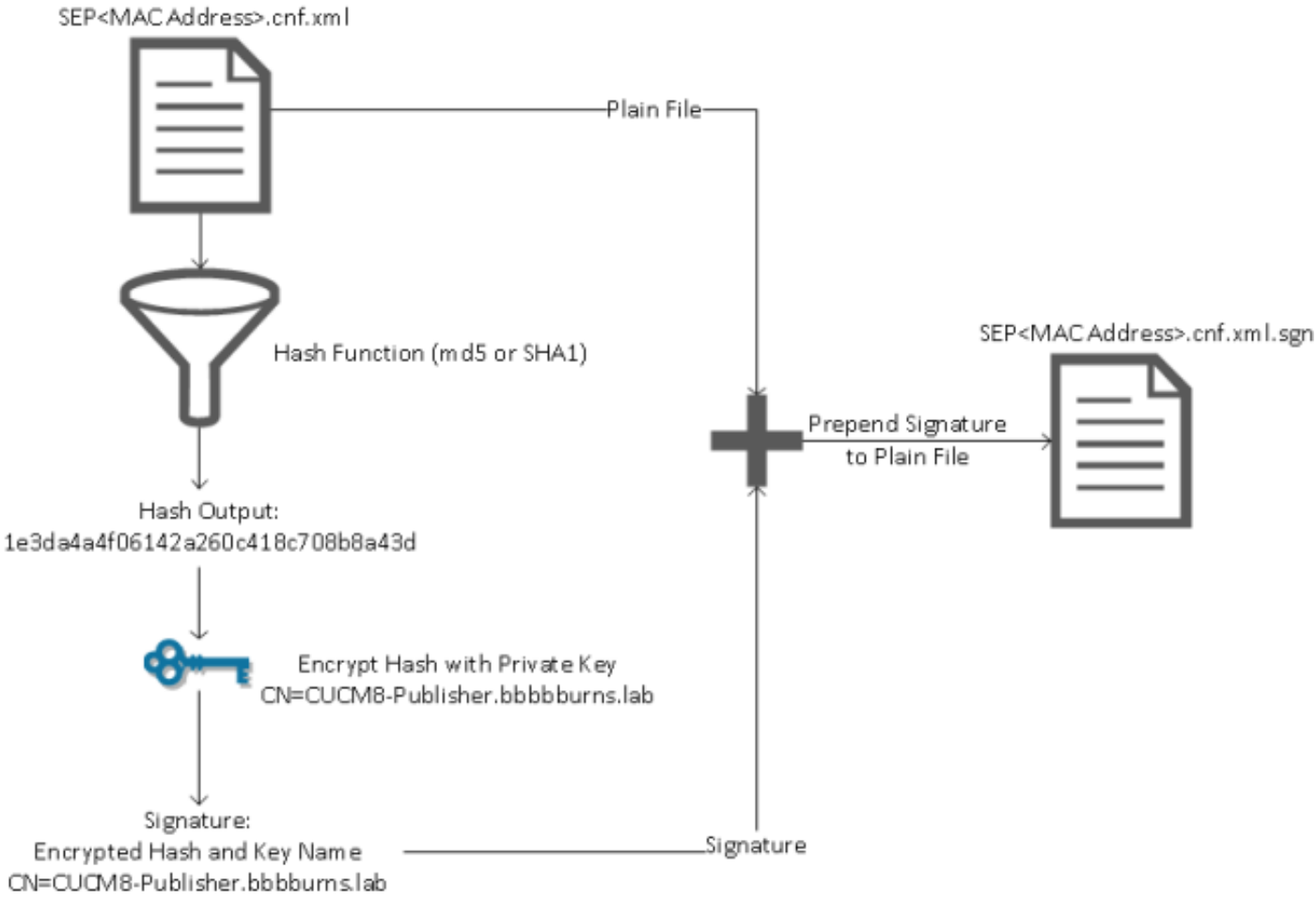
O telefone solicita **SEP *<MAC Address>*.cnf.xml.sgn** para receber o arquivo de configuração com a assinatura especial. Esse arquivo de configuração é assinado pela chave privada TFTP que corresponde ao CallManager.pem na página Gerenciamento de certificado administrativo do sistema operacional (SO).



O arquivo assinado tem uma assinatura na parte superior para autenticar o arquivo, mas está em XML de texto simples. A imagem abaixo mostra que o sinalizador do arquivo de configuração é **CN=CUCM8-Publisher.bbbburns.lab**, por sua vez assinado por **CN=JASBURNS-AD**. Isso significa que o telefone precisa verificar a assinatura do **CUCM8-Publisher.bbbburns.lab** em relação ao arquivo ITL antes que esse arquivo de configuração seja aceito.

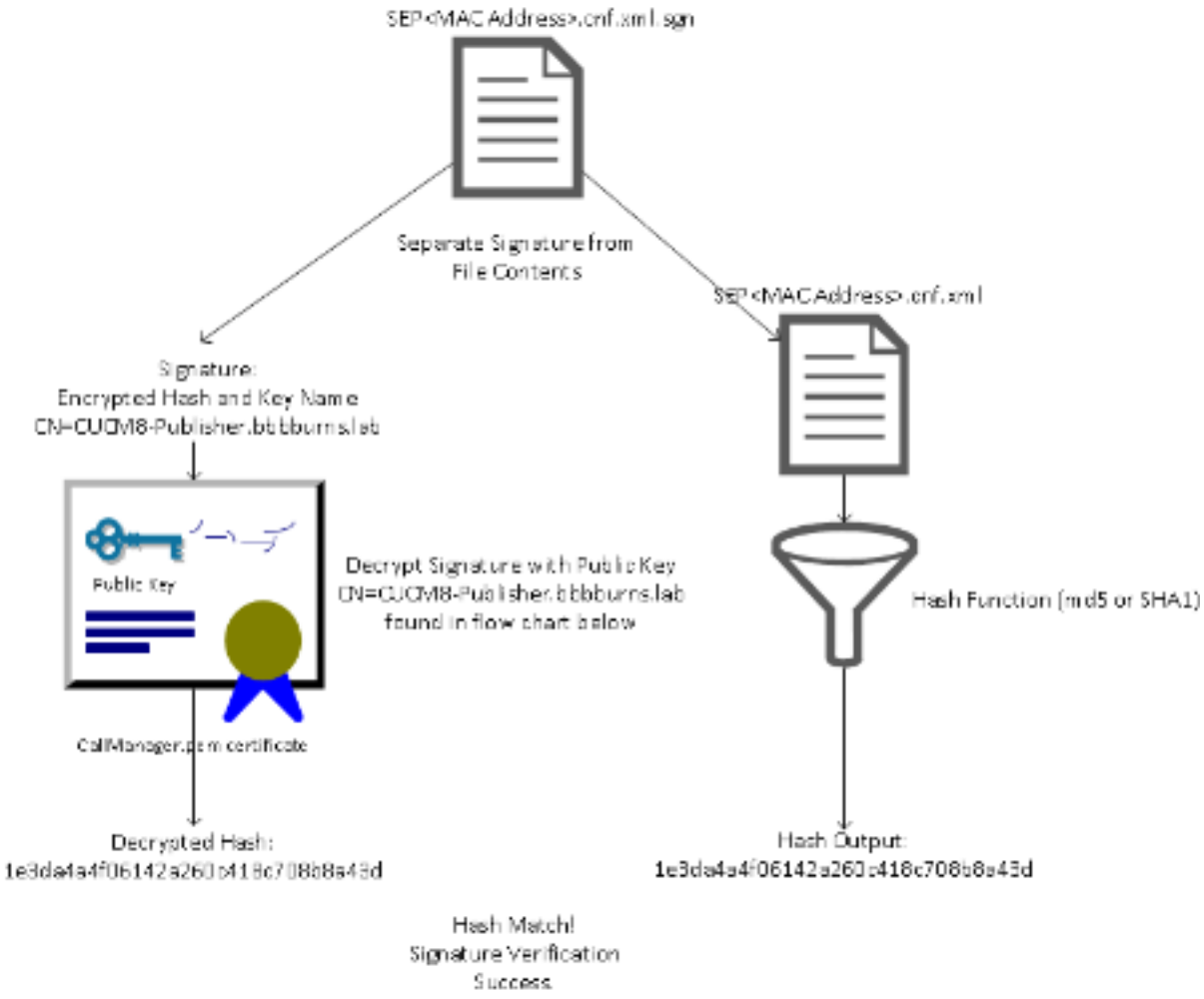


Este é um diagrama que mostra como a chave privada é usada junto com uma função hash Algorithm (MD)5 do Resumo da Mensagem ou Algoritmo de Hash Seguro (SHA)1 para criar o arquivo assinado.



A verificação de assinatura inverte esse processo usando a chave pública correspondente para descriptografar o hash. Se os hashes coincidirem, ele mostrará:

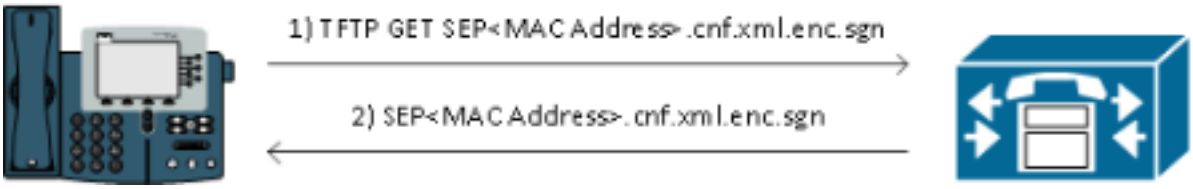
- Este arquivo não foi modificado em trânsito.
- Este arquivo vem da parte listada na assinatura, já que qualquer item descriptografado com êxito com a chave pública deve ter sido criptografado com a chave privada.



Criptografia de arquivo de configuração TFTP

Se a criptografia de configuração TFTP opcional estiver habilitada no Perfil de segurança do telefone associado, o telefone solicitará um arquivo de configuração criptografado. Esse arquivo é assinado com a chave privada TFTP e criptografado com uma chave simétrica trocada entre o telefone e o CUCM (consulte o Cisco Unified Communications Manager Security Guide, Release 8.5(1) para obter todos os detalhes) para que seu conteúdo não possa ser lido com um sniffer de rede a menos que o observador tenha as chaves necessárias.

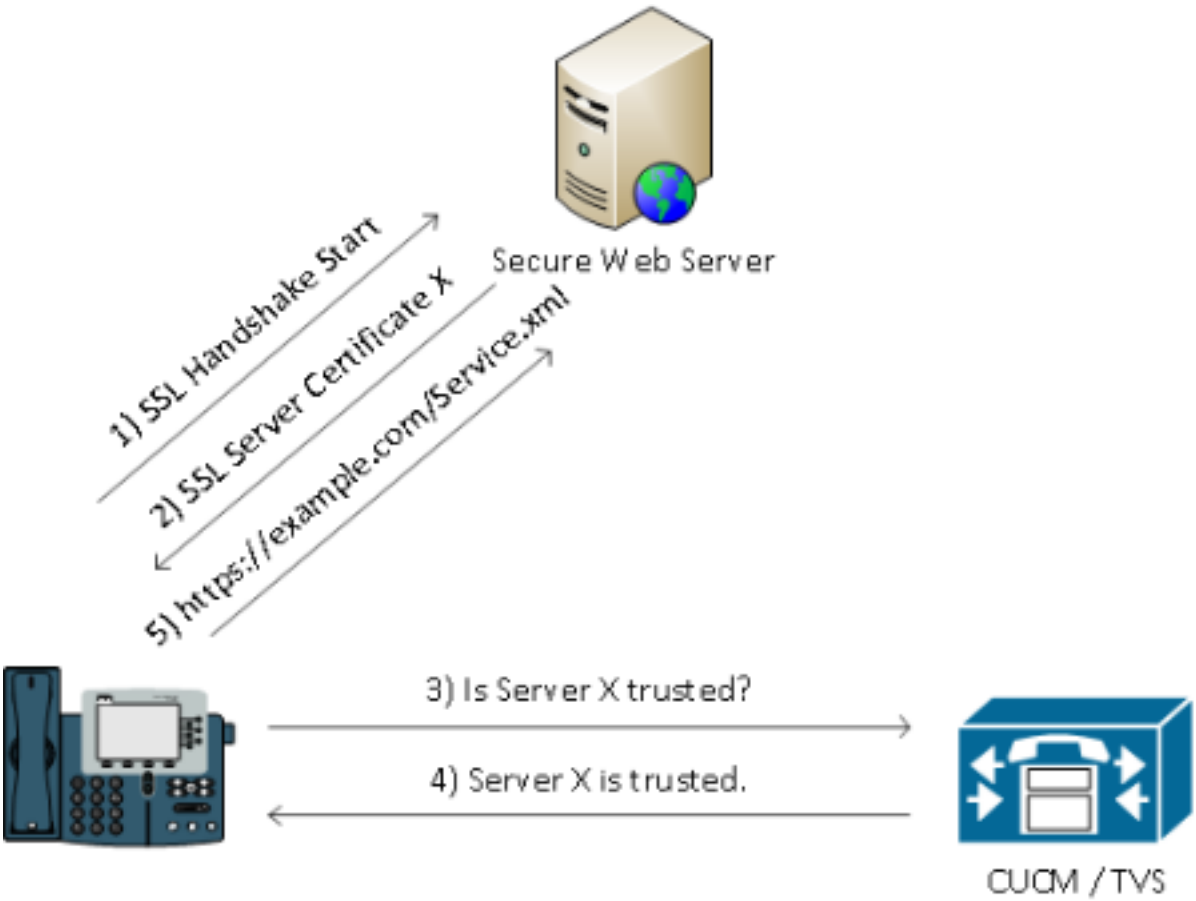
O telefone solicita **SEP *<MAC Address>*.cnf.xml.enc.sgn** para obter o arquivo criptografado assinado.



O arquivo de configuração criptografado também tem a assinatura no início, mas não há dados de texto simples depois, somente dados criptografados (caracteres binários distorcidos neste editor de texto). A imagem mostra que o sinalizador é o mesmo que no exemplo anterior, portanto, esse sinalizador deve estar presente no arquivo ITL antes que o telefone aceite o arquivo. Além disso, as chaves de descriptografia devem estar corretas antes que o telefone possa ler o conteúdo do arquivo.

Serviço de verificação de confiança (verificação remota de certificado e assinatura)

Os telefones IP contêm uma quantidade limitada de memória e também pode haver um grande número de telefones para gerenciar em uma rede. O CUCM atua como um armazenamento de confiança remoto via TVS para que um armazenamento de confiança de certificado completo não tenha que ser colocado em cada telefone IP. Sempre que o telefone não puder verificar uma assinatura ou certificado através dos arquivos CTL ou ITL, ele solicitará a verificação ao servidor TVS. Esse armazenamento confiável central é mais fácil de gerenciar do que se o armazenamento confiável estivesse presente em todos os telefones IP.



Detalhes de SBD e informações sobre solução de problemas

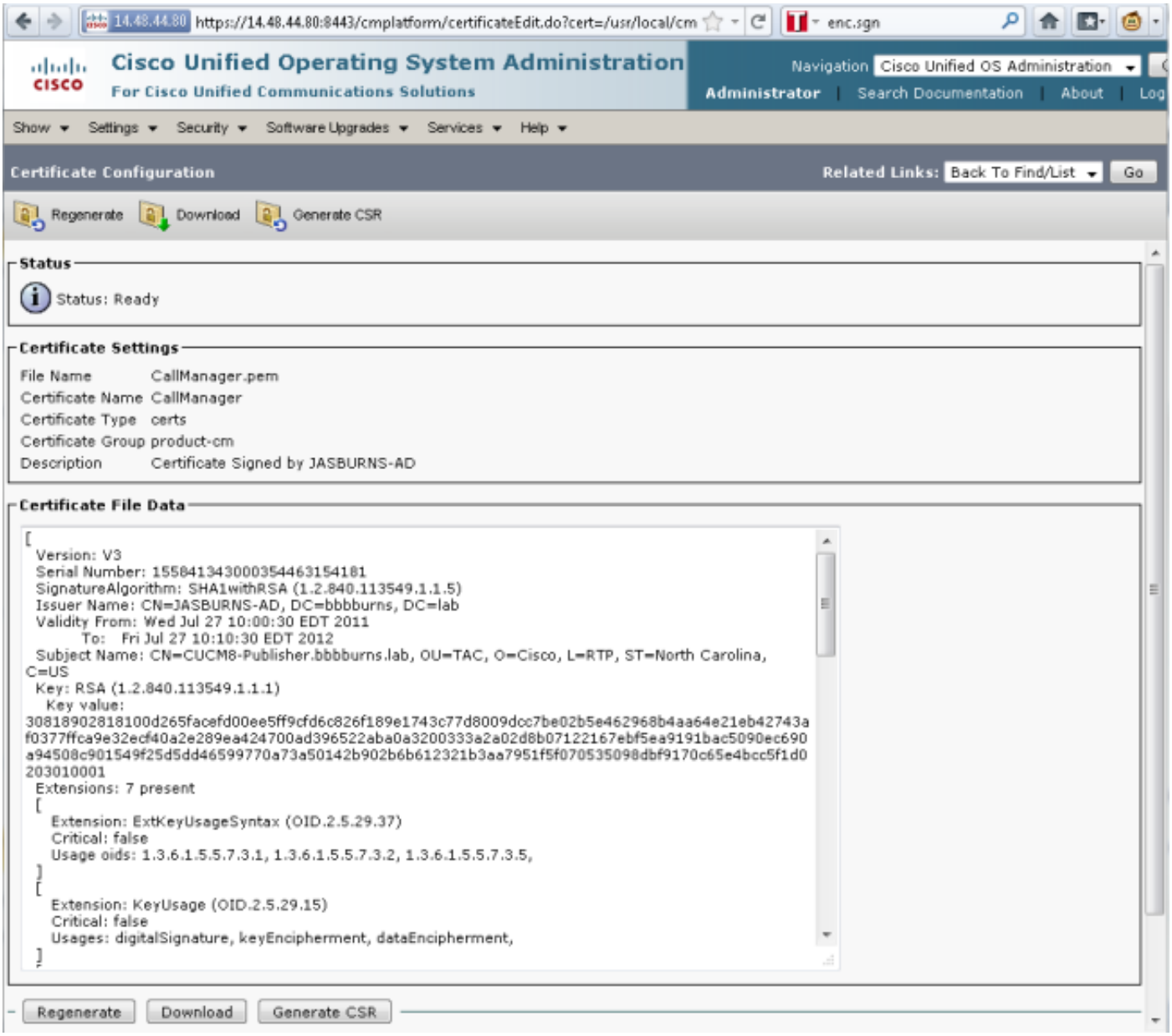
Esta seção detalha o processo SBD.

Arquivos e certificados ITL presentes no CUCM

Primeiro, há vários arquivos que devem estar presentes no próprio servidor CUCM. A peça mais importante é o certificado TFTP e a chave privada TFTP. O certificado TFTP está localizado em **OS Administration > Security > Certificate Management > CallManager.pem**.

O servidor CUCM usa as chaves públicas e privadas do certificado CallManager.pem para o serviço TFTP (bem como para o serviço Cisco Call Manager (CCM)). A imagem mostra que o certificado CallManager.pem é emitido para **CUCM8-publisher.bbbburns.lab** e assinado por **JASBURNS-AD**. Todos os arquivos de configuração TFTP são assinados pela chave privada abaixo.

Todos os telefones podem usar a chave pública TFTP no certificado CallManager.pem para descriptografar qualquer arquivo criptografado com a chave privada TFTP, bem como verificar qualquer arquivo assinado com a chave privada TFTP.



Além da chave privada do certificado CallManager.pem, o servidor CUCM também armazena um arquivo ITL que é apresentado aos telefones. O comando **show itl** mostra o conteúdo completo desse arquivo ITL por meio do acesso Secure Shell (SSH) à CLI do SO do servidor CUCM.

Esta seção detalha o arquivo ITL, peça por peça, porque tem vários componentes importantes que o telefone usa.

A primeira parte são as informações de assinatura. Mesmo o arquivo ITL é um arquivo assinado. Esta saída mostra que é assinada pela chave privada TFTP associada ao certificado anterior do CallManager.pem.

```
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*
```

As próximas seções contêm cada uma sua finalidade dentro de um parâmetro especial **Function**. A primeira função é o System Administrator Security Token. Esta é a assinatura da chave pública TFTP.

```
        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.
```

A próxima função é CCM+TFTP. Esta é novamente a chave pública TFTP que serve para autenticar e descriptografar arquivos de configuração TFTP baixados.

```
        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)
```

A próxima função é TVS. Há uma entrada para a chave pública de cada servidor TVS ao qual o telefone se conecta. Isso permite que o telefone estabeleça uma sessão SSL (Secure Sockets Layer) para o servidor TVS.

```
        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
```

A função final incluída no arquivo ITL é a função de proxy da autoridade de certificação (CAPF). Este certificado permite que os telefones estabeleçam uma conexão segura com o serviço CAPF no servidor CUCM para que o telefone possa instalar ou atualizar um LSC (Locally Significant Certificate). Esse processo será abordado em outro documento que ainda não foi lançado.

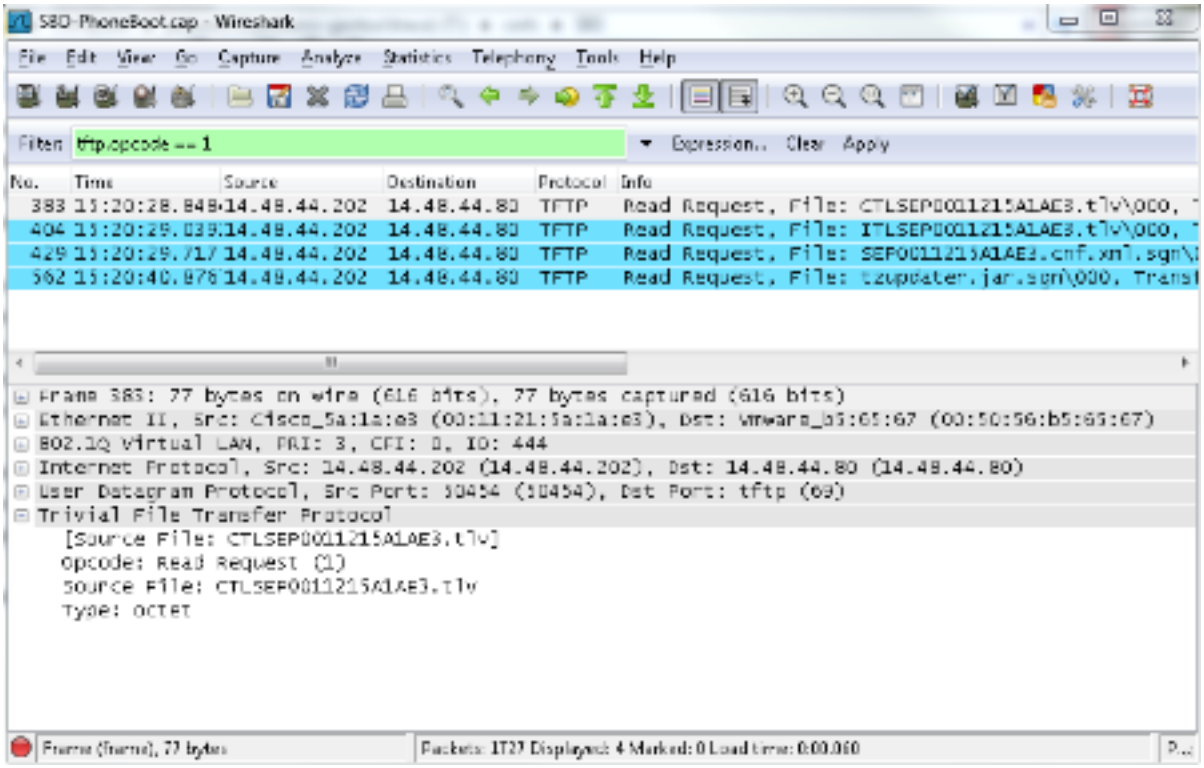
```
        ITL Record #:4
                  ----
BYTEPOS TAG             LENGTH  VALUE
------- ---             ------  -----
1       RECORDLENGTH    2       455
2       DNSNAME         2
3       SUBJECTNAME     61      CN=CAPF-9c4cba7d;
                                OU=TAC;O=Cisco;L=RTP;ST=North Carolina;C=US
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.
```

A próxima seção aborda exatamente o que acontece quando um telefone é inicializado.

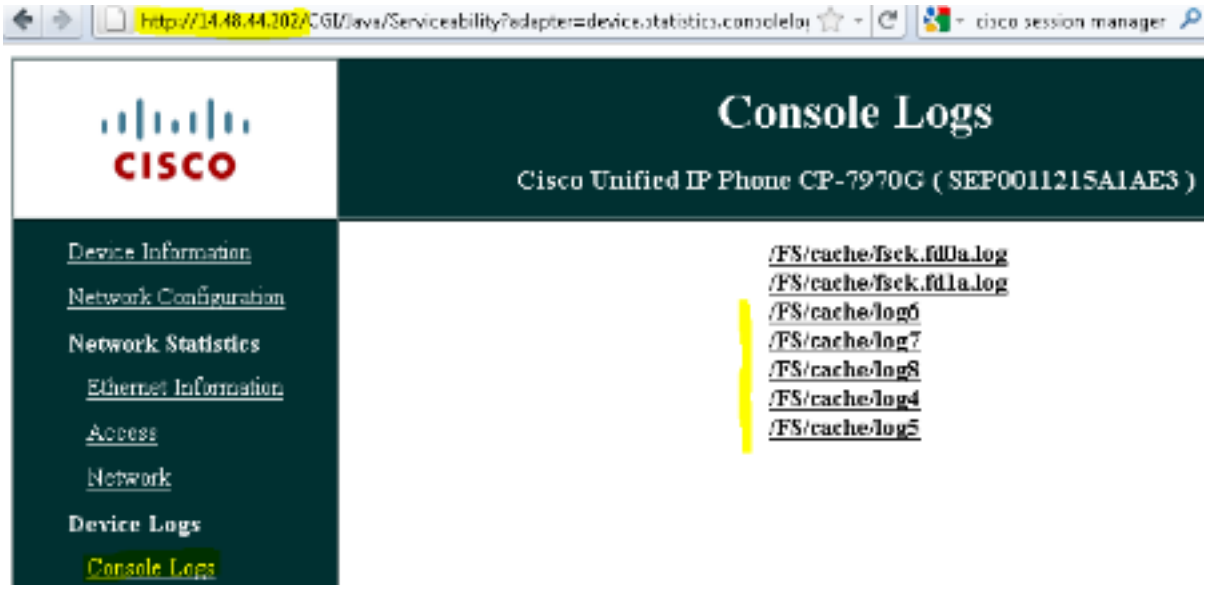
O telefone baixa o ITL e o arquivo de configuração

Depois que o telefone é inicializado e obtém um endereço IP, bem como o endereço de um servidor TFTP, ele solicita os arquivos CTL e ITL primeiro.

Esta captura de pacote mostra uma solicitação de telefone para o arquivo ITL. Se você filtrar em **tftp.opcode == 1,** verá todas as Solicitações de Leitura TFTP do telefone:



Como o telefone recebeu arquivos CTL e ITL do TFTP com êxito, o telefone solicita um arquivo de configuração assinado. Os registros do console do telefone que mostram esse comportamento estão disponíveis na interface da Web do telefone:



Primeiro, o telefone solicita um arquivo CTL, que é bem-sucedido:

```
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
```

Em seguida, o telefone também solicita um arquivo ITL:

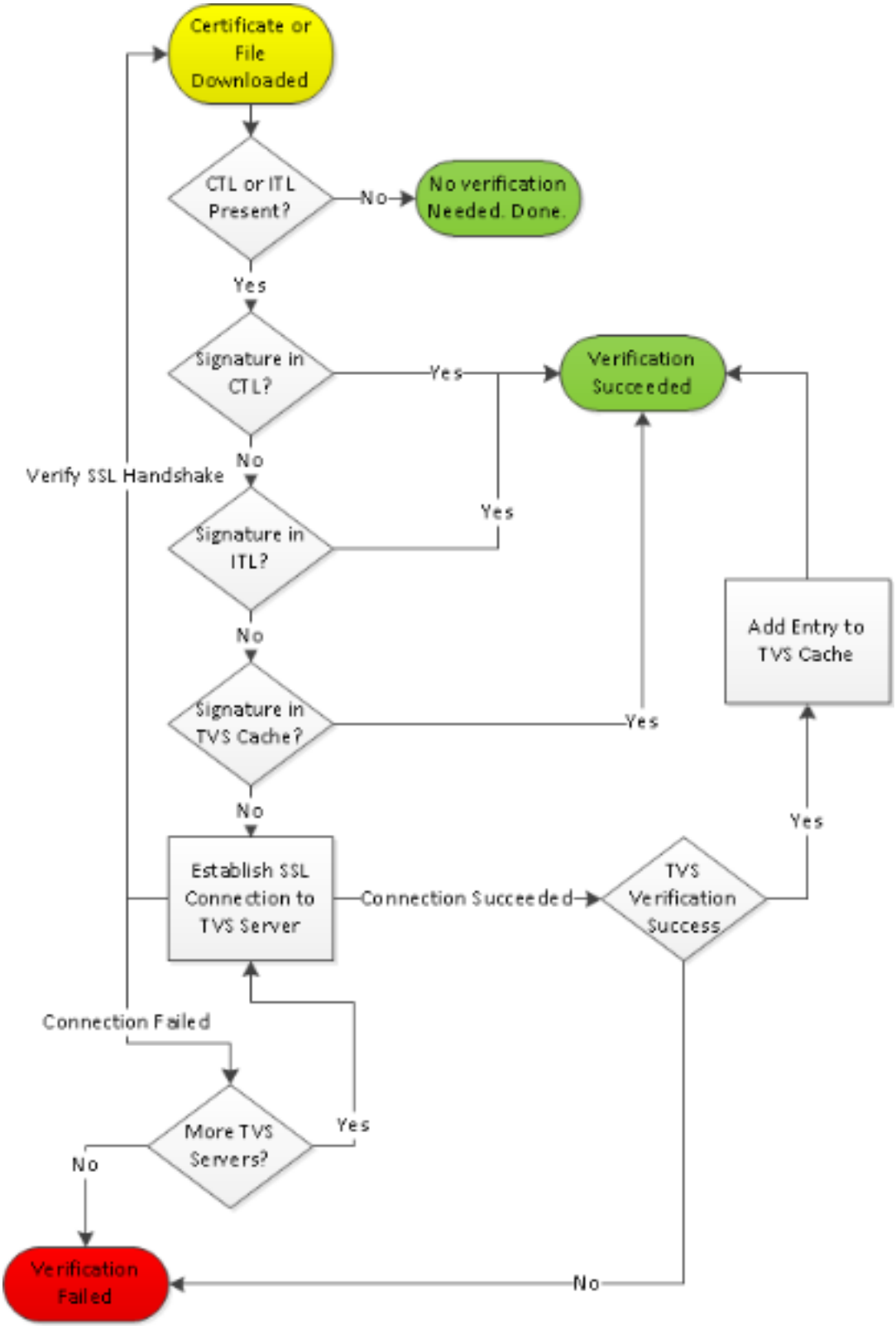
```
868: NOT 09:13:17.860613 TFTP: [28]:Requesting ITLSEP0011215A1AE3.tlv from
14.48.44.80
869: NOT 09:13:17.875059 TFTP: [28]:Finished --> rcvd 5438 bytes
```

O telefone verifica o ITL e o arquivo de configuração

Depois que o arquivo ITL é baixado, ele deve ser verificado. Há vários estados em que um telefone pode estar no momento, portanto este documento cobre todos eles.

- O telefone não tem nenhum arquivo CTL ou ITL presente ou o ITL está em branco devido ao parâmetro **Prepare Cluster para Reversão para Pre 8.0**. neste estado, o telefone confia cegamente no próximo arquivo CTL ou ITL baixado e usa essa assinatura daqui em diante.
- O telefone já tem um CTL, mas não um ITL. Nesse estado, o telefone só confia em um ITL se puder ser verificado pela função CCM+TFTP no arquivo CTL.
- O telefone já tem um CTL e um arquivo ITL. Nesse estado, o telefone verifica se os arquivos baixados recentemente correspondem à assinatura no servidor CTL, ITL ou TVS.

Aqui está um fluxograma que descreve como o telefone verifica os arquivos assinados e os certificados HTTPS:



Nesse caso, o telefone pode verificar a assinatura nos arquivos ITL e CTL. O telefone já tem um CTL e um ITL, então ele simplesmente verificou e encontrou a assinatura correta.

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

Como o telefone baixou os arquivos CTL e ITL, a partir desse ponto, ele solicita SOMENTE arquivos de configuração assinados. Isso ilustra que a lógica do telefone é determinar que o servidor TFTP é seguro, com base na presença de CTL e ITL, e depois solicitar um arquivo assinado:

```
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
```

Depois que o arquivo de configuração assinado é baixado, o telefone deve autenticá-lo em função da função CCM+TFTP dentro do ITL:

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

Contatos telefônicos TVS para certificado desconhecido

O arquivo ITL fornece uma função TVS que contém o certificado do serviço TVS executado na porta TCP 2445 do servidor CUCM. O TVS é executado em todos os servidores em que o serviço CallManager é ativado. O serviço TFTP do CUCM usa o grupo configurado do CallManager para criar uma lista de servidores TVS que o telefone deve contatar no arquivo de configuração do telefone.

Alguns laboratórios usam apenas um único servidor CUCM. Em um cluster CUCM de vários nós, pode haver até três entradas TVS para um telefone, uma para cada CUCM no Grupo CUCM do telefone.

Este exemplo mostra o que acontece quando o botão **Diretórios** no telefone IP é pressionado. A URL de diretórios é configurada para HTTPS, de modo que o telefone é apresentado com o certificado Web Tomcat do servidor Diretórios. Este certificado da Web Tomcat (tomcat.pem na Administração do SO) não está carregado no telefone, portanto, o telefone deve entrar em contato com o TVS para autenticar o certificado.

Consulte o diagrama anterior da Visão geral do TVS para obter uma descrição da interação. Esta é a perspectiva do registro do console do telefone:

Primeiro você encontra o URL do diretório:

```
1184: NOT 15:20:55.219275 JVM: Startup Module Loader|cip.dir.TandunDirectories:
? - Directory url https://14.48.44.80:8443/ccmcip/xmldirectory.jsp
```

Esta é uma sessão HTTP segura SSL/Transport Layer Security (TLS) que requer verificação.

```
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
```

O telefone primeiro verifica se o certificado apresentado pelo servidor SSL/TLS está presente no CTL. Em seguida, o telefone examina as Funções no arquivo ITL para ver se encontra uma correspondência. Essa mensagem de erro diz "certificado HTTPS não em CTL", o que significa "que a certificação não pode ser encontrada no CTL ou no ITL".

```
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>
```

Depois que o conteúdo direto dos arquivos CTL e ITL for verificado quanto ao certificado, a próxima verificação do telefone será o cache TVS. Isso é feito para reduzir o tráfego de rede se o telefone tiver solicitado recentemente o mesmo certificado ao servidor TVS. Se o certificado HTTPS não for encontrado no cache do telefone, você poderá fazer uma conexão TCP com o próprio servidor TVS.

```
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.
```

Lembre-se de que a conexão com o próprio TVS é SSL/TLS (HTTP seguro ou HTTPS), portanto, também é um certificado que precisa ser autenticado em relação ao CTL para ITL. Se tudo correr bem, o certificado do servidor TVS deve ser encontrado na função TVS do arquivo ITL. Veja o registro ITL nº 3 no exemplo anterior do arquivo ITL.

```
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>
```

Sucesso! O telefone agora tem uma conexão segura com o servidor TVS. A próxima etapa é perguntar ao servidor TVS "Olá, confio neste certificado de servidor de diretórios?"

Este exemplo mostra a resposta para essa pergunta - uma resposta de 0 que significa sucesso (sem erro).

```
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
```

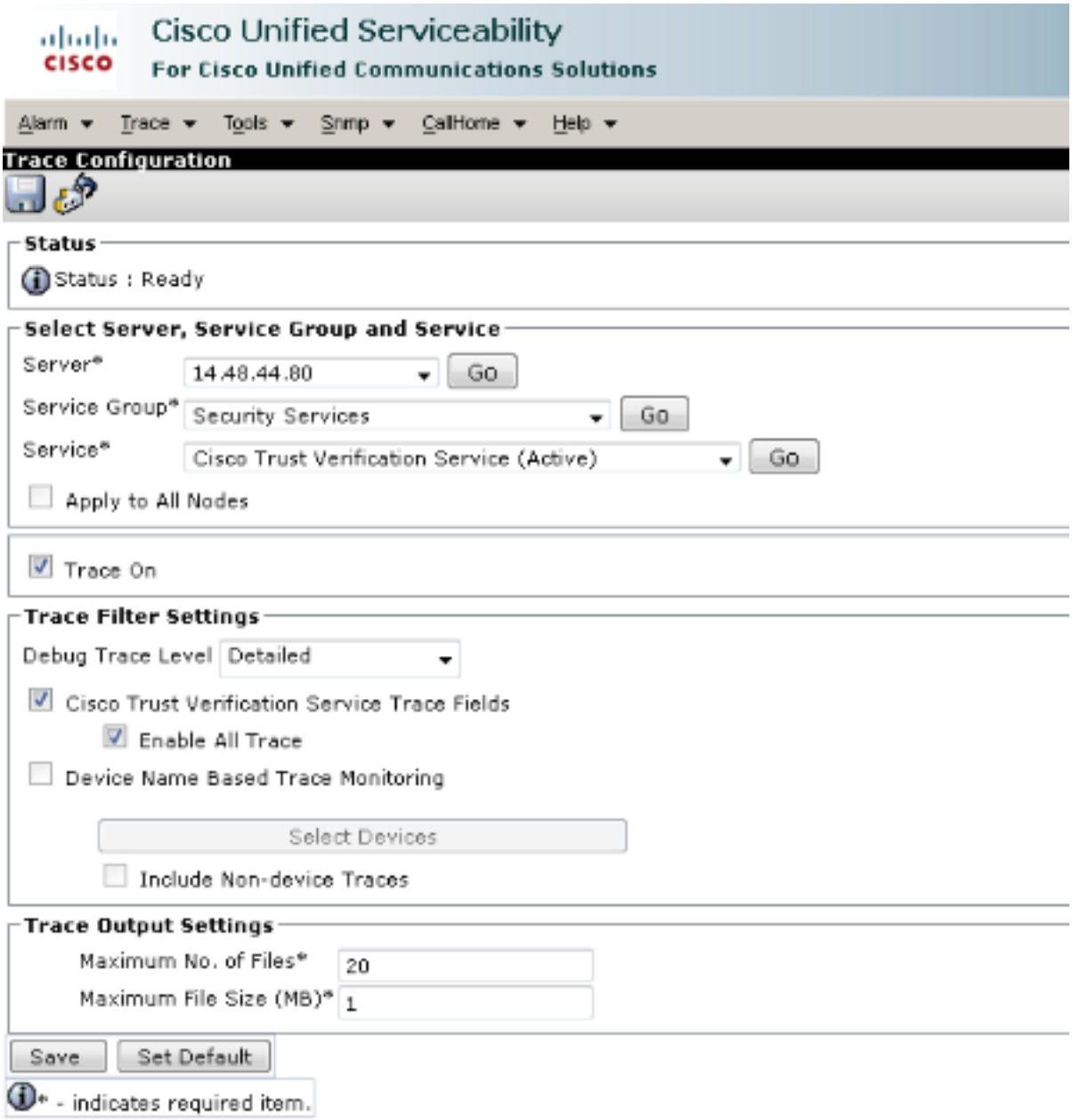
Como há uma resposta bem-sucedida do TVS, os resultados desse certificado são salvos no cache. Isso significa que, se você pressionar o botão **Diretórios** novamente nos próximos 86.400 segundos, não será necessário entrar em contato com o servidor TVS para verificar o certificado. Você pode simplesmente acessar o cache local.

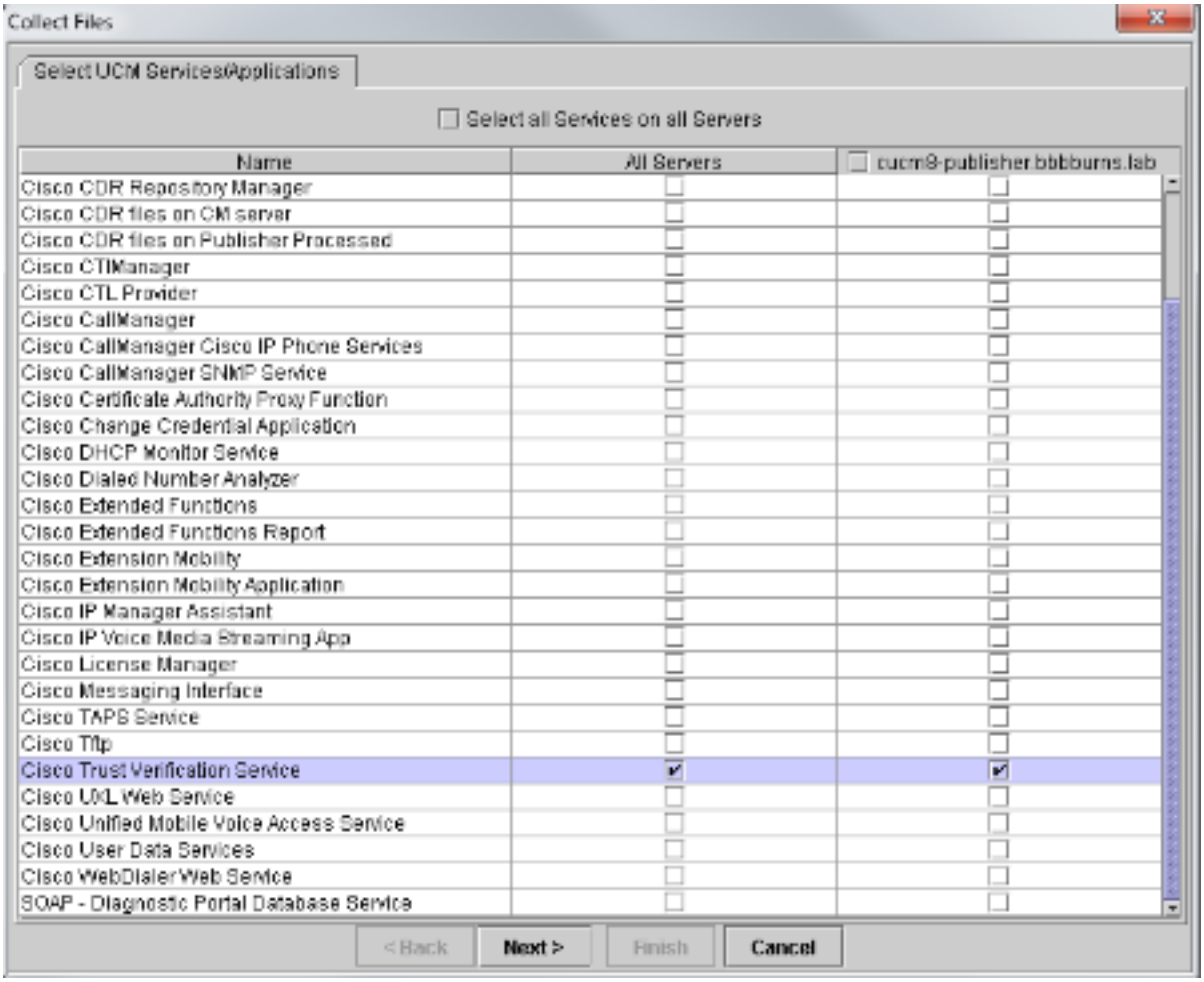
```
1279: NOT 15:20:59.837086 SECD: saveCertToTvsCache: Saving certificate
in TVS cache with default time-to-live value: 86400 seconds
1287: ERR 15:20:59.859993 SECD: Authenticated the HTTPS conn via TVS
```

Finalmente, você verifica se a conexão com o servidor Diretórios foi bem-sucedida.

```
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
```

Aqui está um exemplo do que acontece no servidor CUCM onde o TVS é executado. Você pode coletar registros TVS com a Ferramenta de monitoramento em tempo real (RTMT) do Cisco Unified.


Cisco Unified Serviceability
For Cisco Unified Communications Solutions
Alarm
Trace
Tools
Snmp
CallHome
Help
Trace Configuration
Status
Status : Ready
Select Server, Service Group and Service
Server*
14.48.44.80
Go
Service Group*
Security Services
Go
Service*
Cisco Trust Verification Service (Active)
Go
Apply to All Nodes
Trace On
Trace Filter Settings
Debug Trace Level
Detailed
Cisco Trust Verification Service Trace Fields
Enable All Trace
Device Name Based Trace Monitoring
Select Devices
Include Non-device Traces
Trace Output Settings
Maximum No. of Files*
20
Maximum File Size (MB)*
1
Save
Set Default
* - indicates required item.



Os registros TVS do CUCM mostram que você aperta o handshake SSL com o telefone, o telefone pergunta ao TVS sobre o certificado Tomcat e o TVS responde para indicar que o certificado corresponde ao armazenamento do certificado TVS.

```
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
```

O repositório de certificados TVS é uma lista de todos os certificados contidos na página **da Web Administração do SO > Gerenciamento de certificados**.

Verifique manualmente se o telefone ITL corresponde ao CUCM ITL

Uma concepção equivocada comum observada durante a solução de problemas diz respeito à tendência de excluir o arquivo ITL com a esperança de que ele resolva um problema de verificação de arquivos. Às vezes, a exclusão do arquivo ITL é necessária, mas pode haver uma maneira melhor.

O arquivo ITL só precisa ser excluído quando TODAS essas condições forem atendidas.

- A assinatura do arquivo ITL no telefone não corresponde à assinatura do arquivo ITL no servidor CM TFTP.
- A assinatura do TVS no arquivo ITL não corresponde ao certificado apresentado pelo TVS.
- O telefone mostra "Falha na verificação" quando tenta baixar o arquivo ITL ou os arquivos de configuração.
- Não existe backup da chave privada TFTP antiga.

Aqui está como você verifica as duas primeiras condições.

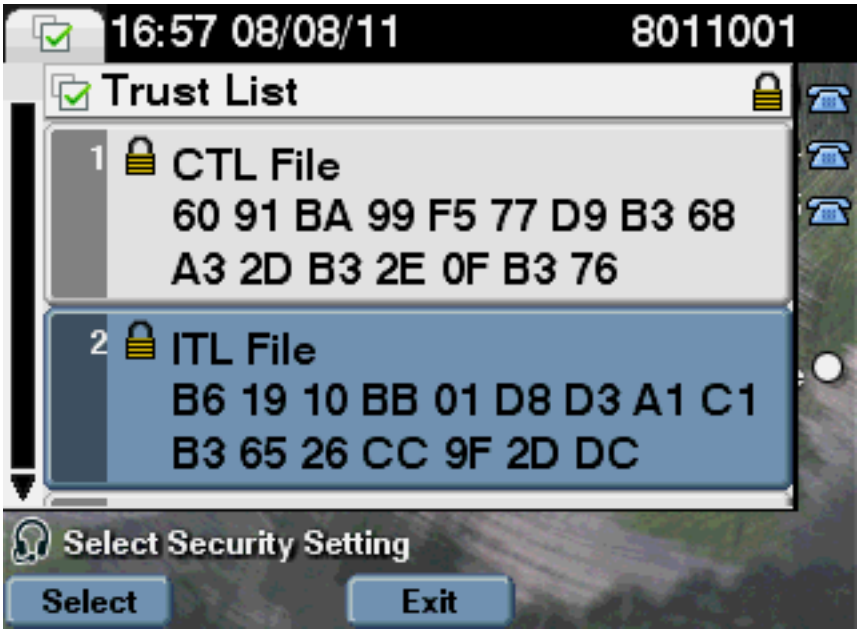
Primeiro, você pode comparar o checksum do arquivo ITL presente no CUCM com o arquivo ITL de checksum do telefone. No momento, não há como examinar a soma MD5do arquivo ITL no CUCM do próprio CUCM até que você execute uma versão com a correção para esse bug da Cisco ID CSCto60209.

Nesse ínterim, execute isso com sua GUI ou programas CLI favoritos:

```
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
```

Isso mostra que o MD5sum do arquivo ITL no CUCM é **b61910bb01d8d3a1c1b36526cc9f2ddc**.

Agora você pode examinar o próprio telefone para determinar o hash do arquivo ITL carregado ali: **Configurações > Configuração de segurança > Lista de confiança**.



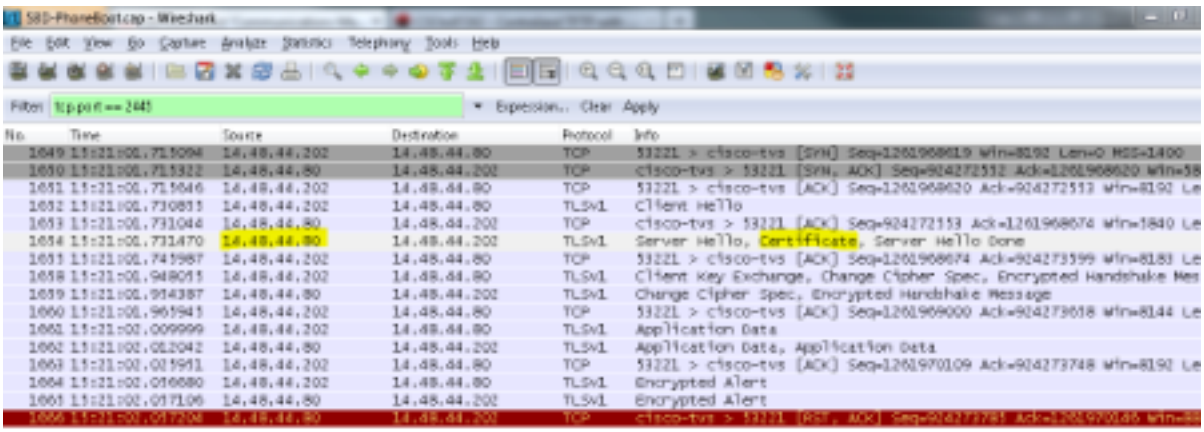
Isso mostra que as somas MD5correspondem. Isso significa que o arquivo ITL no telefone corresponde ao arquivo no CUCM, portanto ele não precisa ser excluído.

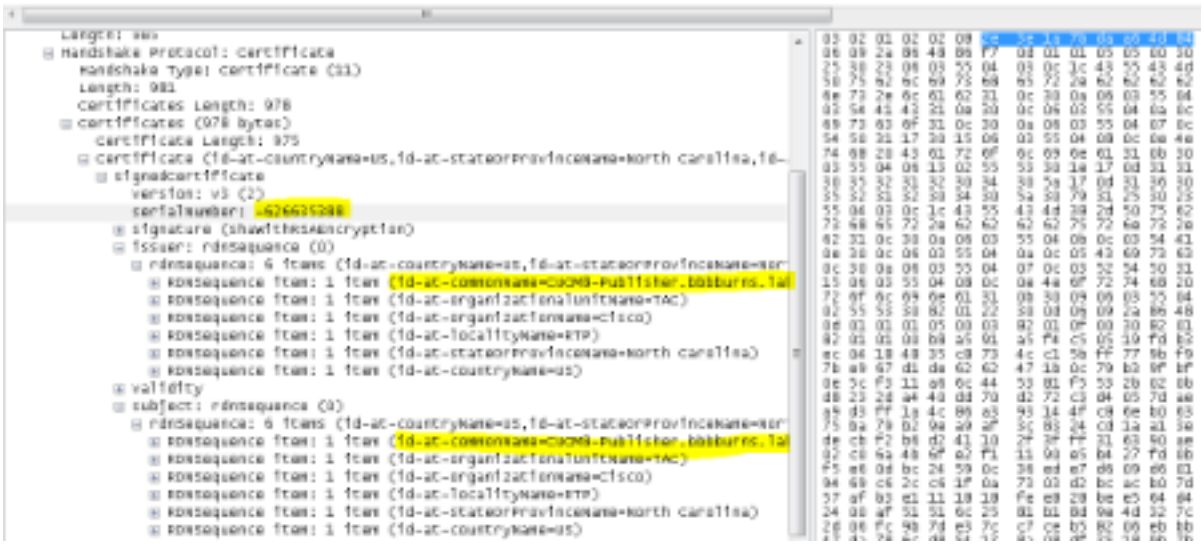
Se ela corresponder, você precisará passar para a próxima operação - determinar se o certificado TVS no ITL corresponde ou não ao certificado apresentado pelo TVS. Esta operação está um pouco mais envolvida.

Primeiro, observe a captura de pacotes do telefone que se conecta ao servidor TVS na porta TCP

2445.

Clique com o botão direito do mouse em qualquer pacote neste fluxo no Wireshark, clique em **Decodificar como** e selecione **SSL**. Localize o certificado do servidor com este aspecto:





Examine o certificado TVS contido no arquivo ITL anterior. Você deve ver uma entrada com o número de série **2E3E1A7BDAA64D84**.

```
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
```

Êxito, o **TVS.pem** dentro do arquivo ITL corresponde ao certificado TVS apresentado na rede. Você não precisa excluir o ITL, e o TVS apresenta o certificado correto.

Se a autenticação de arquivo ainda falhar, verifique o restante do fluxograma anterior.

Restrições e interações

Regenerar certificados / Reconstruir um cluster / Expiração do certificado

O certificado mais importante agora é o certificado CallManager.pem. A chave privada deste certificado é usada para assinar todos os arquivos de configuração TFTP, que incluem o arquivo ITL.

Se o arquivo CallManager.pem for gerado novamente, um novo certificado CCM+TFTP será gerado com uma nova chave privada. Além disso, o arquivo ITL agora é assinado por essa nova chave CCM+TFTP.

Depois de regenerar o CallManager.pem e reiniciar o serviço TVS e TFTP, isso acontece quando um telefone é inicializado.

1. O telefone tenta baixar o novo arquivo ITL assinado pelo novo CCM+TFTP do servidor TFTP. O telefone tem apenas o arquivo ITL antigo neste ponto, e as novas chaves não estão no arquivo ITL presente no telefone.
2. Como o telefone não pôde localizar a nova assinatura CCM+TFTP no ITL antigo, ele tenta entrar em contato com o serviço TVS.
   Note: Esta parte é extremamente importante. O certificado TVS do arquivo ITL antigo ainda deve corresponder. Se o CallManager.pem e o TVS.pem forem regenerados exatamente ao mesmo tempo, os telefones não poderão baixar nenhum arquivo novo sem excluir o ITL do telefone manualmente.
3. Quando o telefone entra em contato com o TVS, o servidor CUCM que executa o TVS tem o novo certificado CallManager.pem no Repositório de Certificados do SO.
4. O servidor TVS retorna bem-sucedido e o telefone carrega o novo arquivo ITL na memória.
5. O telefone agora tenta baixar um arquivo de configuração que foi assinado pela nova chave CallManager.pem.
6. Como o novo ITL foi carregado, o arquivo de configuração recém-assinado é verificado com êxito pelo ITL na memória.

Pontos principais:

- Nunca gere novamente os certificados CallManager.pem e TVS.pem ao mesmo tempo.
- Se TVS.pem ou CallManager.pem forem regenerados, TVS e TFTP devem ser reiniciados e os telefones redefinidos para obter os novos arquivos ITL. As versões mais recentes do CUCM lidam com essa redefinição de telefone automaticamente e avisam o usuário no momento da regeneração do certificado.
- Se existir mais de um servidor TVS (mais de um servidor no Grupo CallManager), os servidores adicionais poderão autenticar o novo certificado CallManager.pem.

Mover telefones entre clusters

Quando você move telefones de um cluster para outro com ITLs instalados, a ITL e a chave privada TFTP devem ser levadas em conta. Qualquer novo arquivo de configuração apresentado ao telefone DEVE corresponder a uma assinatura em CTL, ITL ou a uma assinatura no serviço

TVS atual do telefone.

Este documento explica como garantir que o arquivo ITL e os arquivos de configuração do novo cluster possam ser confiáveis pelo arquivo ITL atual no telefone. https://supportforums.cisco.com/docs/DOC-15799.

Backup E Restauração

O backup do certificado e da chave privada do CallManager.pem é feito por meio do Sistema de Recuperação de Desastre (DRS). Se um servidor TFTP for recriado, ele DEVE ser restaurado do backup para que a chave privada possa ser restaurada. Sem a chave privada CallManager.pem no servidor, os telefones com ITLs atuais que usam a chave antiga não confiam em arquivos de configuração assinados.

Se um cluster for reconstruído e não restaurado a partir do backup, ele será exatamente como o documento "Movendo telefones entre clusters". Isso porque um cluster com uma nova chave é um cluster diferente no que diz respeito aos telefones.

Há um defeito grave associado ao backup e à restauração. Se um cluster for susceptível ao bug da Cisco ID CSCtn50405, os backups do DRS não conterão o certificado CallManager.pem. Isso faz com que qualquer servidor restaurado desse backup gere arquivos ITL corrompidos até que um novo CallManager.pem seja gerado. Se não houver outros servidores TFTP funcionais que não passaram pela operação de backup e restauração, isso pode significar que todos os arquivos ITL precisam ser excluídos dos telefones.

Para verificar se o arquivo CallManager.pem precisa ser regenerado, insira o comando **show itl** seguido por:

```
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
```

Na saída ITL, os principais erros a serem procurados são:

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

e

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

A consulta SQL (Structured Query Language) anterior procura os certificados que têm uma função de "Autenticação e Autorização". O certificado CallManager.pem na consulta de banco de dados anterior que tem a função de Autenticação e Autorização TAMBÉM deve estar presente na página da Web Gerenciamento de Certificados da Administração do SO. Se o defeito anterior for encontrado, há uma incompatibilidade entre os certificados CallManager.pem na consulta e na página da Web do SO.

Alterar nomes de host ou nomes de domínio

Se você alterar o nome de host ou o nome de domínio de um servidor CUCM, ele regenera todos

os certificados de uma só vez nesse servidor. A seção de regeneração de certificado explicou que a regeneração do TVS.pem e do CallManager.pem é uma "coisa ruim".

Há alguns cenários em que uma alteração de nome de host falha e alguns em que ela funciona sem problemas. Esta seção abrange todos eles e os vincula de volta ao que você já sabe sobre TVS e ITL deste documento.

Cluster de nó único com apenas ITL (tenha cuidado, isso é interrompido sem preparação)

- Com um servidor Business Edition ou implantação somente de editor, o CallManager.pem e o TVS.pem são regenerados ao mesmo tempo em que você altera os nomes de host.
- Se o nome do host for alterado em um cluster de nó único sem primeiro usar o parâmetro Rollback Enterprise abordado aqui, os telefones não poderão verificar o novo arquivo ITL ou os arquivos de configuração em relação ao seu arquivo ITL atual. Além disso, eles não podem se conectar ao TVS porque o certificado TVS também não é mais confiável.
- Os telefones exibem um erro sobre "Trust List Verification Failed" (Falha na verificação da lista de confiança), nenhuma alteração nova na configuração é aplicada e URLs de serviço seguro falham.
- A única solução se a precaução na etapa 2 não for tomada pela primeira vez é excluir manualmente o ITL de cada telefone.

Cluster de nó único com CTL e ITL (isso pode ser interrompido temporariamente, mas facilmente corrigido)

- Depois de executar a mudança de nome dos servidores, execute novamente o cliente CTL. Isso coloca o novo certificado CallManager.pem no arquivo CTL baixado pelo telefone.
- Novos arquivos de configuração, que incluem os novos arquivos ITL, podem ser confiáveis com base na função CCM+TFTP no arquivo CTL.
- Isso funciona porque o arquivo CTL atualizado é confiável com base em uma chave privada USB eToken que permanece a mesma.

Cluster com vários nós com apenas ITL (geralmente funciona, mas pode ser interrompido permanentemente se feito com pressa)

- Como um cluster de vários nós tem vários servidores TVS, qualquer servidor único pode ter seus certificados regenerados sem um problema. Quando o telefone é apresentado com esta nova assinatura desconhecida, ele pede a outro dos servidores TVS para verificar o novo certificado do servidor.
- Há dois problemas principais que podem fazer com que isso falhe:
  Se todos os servidores forem renomeados e reinicializados ao mesmo tempo, nenhum dos servidores TVS poderá ser alcançado com certificados conhecidos quando os servidores e telefones forem reativados.Se um telefone tiver apenas um único servidor no grupo do CallManager, os servidores TVS adicionais não farão diferença. Consulte o cenário "Cluster de nó único" para resolver isto ou adicione outro servidor ao grupo do CallManager do telefone.

Cluster de vários nós com CTL e ITL (isso não pode ser interrompido permanentemente)

- Depois de executar os renomeações, o serviço TVS autentica os novos certificados.
- Mesmo que todos os servidores TVS estejam indisponíveis por algum motivo, o cliente CTL ainda pode ser usado para atualizar os telefones com os novos certificados CallManager.pem CCM+TFTP.

TFTP centralizado

Quando um telefone com um ITL inicializa, ele solicita estes arquivos: **CTLSEP<MAC Address>.tlv, ITLSEP *<MAC Address>*.tlv** e **SEP*<MAC Address>*.cnf.xml.sgn**.

Se o telefone não puder localizar esses arquivos, ele solicitará o **ITLFile.tlv** e o **CTLFile.tlv**, que um servidor TFTP centralizado fornece a qualquer telefone que o solicite.

Com o TFTP centralizado, há um único cluster TFTP que aponta para vários outros subclusters. Frequentemente isso é feito porque os telefones em vários clusters do CUCM compartilham o mesmo escopo de DHCP e, portanto, devem ter o mesmo servidor TFTP da Opção 150 de DHCP. Todos os telefones IP apontam para o cluster TFTP central, mesmo que se registrem em outros clusters. Este servidor TFTP central consulta os servidores TFTP remotos sempre que recebe uma solicitação de um arquivo que não pode ser encontrado.

Devido a essa operação, o TFTP centralizado só funciona em um ambiente ITL homogêneo. Todos os servidores devem executar o CUCM versão 8.x ou posterior, ou todos os servidores devem executar versões anteriores à versão 8.x.

Se um ITLFile.tlv for apresentado do servidor TFTP centralizado, os telefones não confiarão em nenhum arquivo do servidor TFTP remoto porque as assinaturas não correspondem. Isto acontece numa mistura heterogênea. Em uma combinação homogênea, o telefone solicita **ITLSEP *<MAC>*.tlv** que é extraído do cluster remoto correto.

Em um ambiente heterogêneo com uma combinação de clusters anteriores à versão 8.x e versão 8.x, o "Prepare Cluster para Reversão para Pré 8.0" deve ser ativado no cluster da versão 8.x conforme descrito no bug da Cisco ID CSCto87262 e os "Parâmetros de URL de telefone seguro" configurados com HTTP em vez de HTTPS. Isso efetivamente desabilita as funções ITL no telefone.

Perguntas mais freqüentes

Posso desligar o SBD?

Você só pode desativar o SBD se o SBD e o ITL estiverem funcionando no momento.

O SBD pode ser desativado temporariamente em telefones com o Preparar Cluster para Reversão para Parâmetro Empresarial anterior a 8,0" e configurando os "Parâmetros de URL do Telefone Seguro" com HTTP em vez de HTTPS. Quando você define o parâmetro Rollback, ele cria um arquivo ITL assinado com entradas de função em branco. O arquivo ITL "vazio" ainda está assinado, portanto, o cluster deve estar em um estado de segurança totalmente funcional antes que esse parâmetro possa ser ativado.

Depois que esse parâmetro é ativado e o novo arquivo ITL com entradas em branco é baixado e verificado, os telefones aceitam qualquer arquivo de configuração, independentemente de quem o tenha assinado.

Não é recomendável deixar o cluster nesse estado, pois nenhuma das três funções mencionadas anteriormente (arquivos de configuração autenticados, arquivos de configuração criptografados e

URLs HTTPS) está disponível.

Posso excluir facilmente o arquivo ITL de todos os telefones quando o CallManager.pem for perdido?

Não há nenhum método para excluir todos os ITLs de um telefone fornecido remotamente pela Cisco. É por isso que os procedimentos e interações descritos neste documento são tão importantes a ter em conta.

No momento, há uma melhoria não resolvida da ID de bug Cisco CSCto47052 que solicita essa funcionalidade, mas ainda não foi implementada.

No período intermediário, um novo recurso foi adicionado através do bug da Cisco ID CSCts01319 que pode permitir que o Cisco Technical Assistance Center (TAC) reverta para o ITL anteriormente confiável se ainda estiver disponível no servidor. Isso só funciona em certas instâncias em que o cluster está em uma versão com essa correção de defeito e em que o ITL anterior existe em um backup armazenado em um local especial no servidor. Veja o defeito para ver se sua versão tem a correção. Entre em contato com o Cisco TAC para executar o procedimento de recuperação potencial explicado no defeito.

Se o procedimento anterior não estiver disponível, as teclas do telefone deverão ser pressionadas manualmente no telefone para excluir o arquivo ITL. Esta é a compensação feita entre segurança e facilidade de administração. Para que o arquivo ITL seja realmente seguro, ele não deve ser removido remotamente com facilidade.

Mesmo com botões com script pressionando com objetos XML SOAP (Simple Object Access Protocol), o ITL não pode ser remotamente removido. Isso ocorre porque, neste momento, o acesso TVS (e, portanto, o acesso ao URL de autenticação segura para validar objetos recebidos por botão XML SOAP) não está funcionando. Se a URL de autenticação não estiver configurada como segura, talvez seja possível fazer o script das teclas para excluir um ITL, mas esse script não está disponível na Cisco.

Outros métodos para script de teclas remotas sem usar o URL de autenticação podem estar disponíveis de terceiros, mas esses aplicativos não são fornecidos pela Cisco.

O método usado com mais frequência para excluir o ITL é um broadcast de e-mail para todos os usuários do telefone que os instrui sobre a sequência de teclas. Se as configurações de acesso estiverem definidas como **Restrito** ou **Desativado**, o telefone precisará ser redefinido de fábrica, pois os usuários não terão acesso ao menu Configurações do telefone.