Modo misturado CUCM com Tokenless CTL

Índice

Introdução

Este documento descreve a diferença entre a Segurança do gerente das comunicações unificadas de Cisco (CUCM) com e sem o uso de eTokens do hardware USB. Este documento igualmente descreve as hipóteses de implementação básicas que envolvem o certificate trust list de Tokenless (CTL) e o processo que é usado a fim se assegurar de que as funções de sistema corretamente após as mudanças.

Pré-requisitos

Requisitos

Cisco recomenda que você tem o conhecimento da versão 10.0(1) ou mais recente CUCM. Adicionalmente, assegure isso:

- Você tem o acesso administrativo ao comando line interface(cli) do nó do editor CUCM.
- Você tem o acesso aos eTokens do hardware USB e aquele o cliente CTL de encaixe é instalado em seu PC para as encenações que o exigem migrar de volta ao uso de eTokens do hardware.
- Há uma conectividade direta entre todos os Nós CUCM no conjunto. Isto é muito importante porque o arquivo CTL é copiado a todos os Nós no conjunto através do protocolo de transferência de arquivo SSH (SFTP).
- A replicação do base de dados (DB) no conjunto trabalha corretamente e aquela os server

replicate os dados no tempo real.

- Os dispositivos em seu desenvolvimento apoiam a Segurança à revelia (TV). Você pode usar a *lista unificada dos recursos de telefone CM de* Cisco unificada relatando o Web page (IP ou FQDN>/cucreports/de https:// <CUCM) a fim determinar os dispositivos que apoiam a Segurança à revelia. **Note**: Cisco Jabber e muito os Telefones IP do 7940/7960 Series do Cisco TelePresence ou do Cisco não apoiam atualmente a Segurança à revelia. Se você distribui Tokenless CTL com os dispositivos que não apoiam a Segurança à revelia, toda a atualização a seu sistema que muda o certificado do CallManager no editor impedirá que aqueles dispositivos se registrem com o sistema até que seu CTL esteja suprimido manualmente.

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- Versão 10.5.1.10000-7 CUCM (conjunto de dois Nós)
- Telefones IP do Cisco 7975 Series registrados através do protocolo skinny client control (SCCP) com versão de firmware SCCP75.9-3-1SR4-1S
- Dois tokens do Cisco Security que são usados a fim ajustar o conjunto a modo misturado com o uso do software do cliente CTL

Informações de Apoio

Tokenless CTL é uns novos recursos em versões 10.0(1) e mais recente CUCM que permitam a criptografia da sinalização de chamada e dos media para Telefones IP sem a necessidade de usar eTokens do hardware USB e o cliente CTL de encaixe, que era a exigência em liberações precedentes CUCM.

Quando o conjunto é colocado em modo misturado com o uso do comando CLI, o arquivo CTL está assinado com o certificado CCM+TFTP (server) do nó do editor, e há nenhum eToken os Certificados atuais no arquivo CTL.

> **Note**: Quando você regenera o certificado do CallManager (CCM+TFTP) no editor, muda o signatário do arquivo. Os telefones e os dispositivos que não apoiam a Segurança à revelia não aceitarão o arquivo novo CTL a menos que os arquivos CTL **forem suprimidos manualmente de cada dispositivo**. Refira a última exigência que é alistada a seção das exigências deste documento para mais informação.

Do modo NON-seguro a modo misturado (Tokenless CTL)

Esta seção descreve o processo que é usado a fim mover a Segurança do conjunto CUCM em modo misturado através do CLI.

Antes desta encenação, o CUCM reagia do modo NON-seguro, assim que significa que não havia

nenhum arquivo CTL atual em alguns dos Nós e que os Telefones IP registrados tiveram somente um arquivo da lista da confiança da identidade (ITL) instalado, segundo as indicações destas saídas:

```
admin:show ctl
Length of CTL file: 0
CTL File not found. Please run CTLClient plugin or run the CLI - utils ctl.. to
generate the CTL file.
Error parsing the CTL File.
admin:
```

A fim mover-se os CUCM aglomeram a Segurança em modo misturado com o uso da característica nova de Tokenless CTL, terminam estas etapas:

1. Obtenha o acesso administrativo ao nó CLI do editor CUCM.

2. Incorpore o comando do grupo-**conjunto misturado-MODE do ctl dos utils no** CLI:

   ```
   admin:utils ctl set-cluster mixed-mode
   This operation will set the cluster to Mixed mode. Do you want to continue? (y/n):y

   Moving Cluster to Mixed Mode
   Cluster set to Mixed Mode
   Please Restart the TFTP and Cisco CallManager services on all nodes in the cluster
   that run these services
   admin:
   ```

3. Navegue à **página de admin > ao sistema > parâmetros de empreendimento CUCM** e verifique se o conjunto esteve ajustado a modo misturado (um valor de **1** indica modo misturado):

4. Reinicie o TFTP e os serviços do CallManager da Cisco em todos os Nós no conjunto que dirigem estes serviços.

5. Reinicie todos os Telefones IP de modo que possam obter o arquivo CTL do serviço TFTP CUCM.

6. A fim verificar o índice do arquivo CTL, incorpore o comando do **ctl da mostra no** CLI. No arquivo CTL você pode ver que o certificado CCM+TFTP (server) para o nó do editor CUCM está usado a fim assinar o arquivo CTL (este arquivo é o mesmo em todos os server no conjunto). Está aqui um exemplo de saída:

   ```
   admin:show ctl
   The checksum value of the CTL file:
   0c05655de63fe2a042cf252d96c6d609(MD5)
   8c92d1a569f7263cf4485812366e66e3b503a2f5(SHA1)

   Length of CTL file: 4947
   The CTL File was last modified on Fri Mar 06 19:45:13 CET 2015

   [...]

   CTL Record #:1
   ----
   ```

```
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 System Administrator Security Token
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D 21
A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4
This etoken was used to sign the CTL file.
CTL Record #:2
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 CCM+TFTP
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D 21
A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4

[...]

The CTL file was verified successfully.
```

7. No lado do telefone IP, você pode verificar que depois que o serviço é reiniciado, transfere o arquivo CTL, que está agora atual no servidor TFTP (a soma de verificação MD5 combina quando comparado à saída do CUCM):

   Note: Quando você verifica a soma de verificação no telefone, você vê o **MD5** ou o **SHA1**, dependente do tipo de telefone.

Dos eTokens do hardware à solução de Tokenless

Esta seção descreve como migrar a Segurança do conjunto CUCM dos eTokens do hardware ao uso da solução nova de Tokenless.

Em algumas situações, o modo misturado é configurado já no CUCM com o uso do cliente CTL, e nos arquivos do uso CTL dos Telefones IP que contêm os Certificados dos eTokens do hardware USB. Com esta encenação, o arquivo CTL é assinado por um certificado de um dos eTokens USB e instalado nos Telefones IP. Aqui em um exemplo:

```
admin:show ctl
The checksum value of the CTL file:
256a661f4630cd86ef460db5aad4e91c(MD5)
```

```
3d56cc01476000686f007aac6c278ed9059fc124(SHA1)

Length of CTL file: 5728
The CTL File was last modified on Fri Mar 06 21:48:48 CET 2015

[...]
CTL Record #:5
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1186
2 DNSNAME 1
3 SUBJECTNAME 56 cn="SAST-ADN008580ef ";ou=IPCBU;o="Cisco Systems
4 FUNCTION 2 System Administrator Security Token
5 ISSUERNAME 42 cn=Cisco Manufacturing CA;o=Cisco Systems
6 SERIALNUMBER 10 83:E9:08:00:00:00:55:45:AF:31
7 PUBLICKEY 140
9 CERTIFICATE 902 85 CD 5D AD EA FC 34 B8 3E 2F F2 CB 9C 76 B0 93
3E 8B 3A 4F (SHA1 Hash HEX)
10 IPADDRESS 4
This etoken was used to sign the CTL file.

The CTL file was verified successfully.
```

Termine estas etapas a fim mover a Segurança do conjunto CUCM para o uso de Tokenless CTL:

1. Obtenha o acesso administrativo ao nó CLI do editor CUCM.
2. Inscreva o comando CLI de **CTLFile da atualização do ctl dos utils**:

   ```
   admin:utils ctl update CTLFile
   This operation will update the CTLFile. Do you want to continue? (y/n):y

   Updating CTL file
   CTL file Updated
   Please Restart the TFTP and Cisco CallManager services on all nodes in
   the cluster that run these services
   ```

3. Reinicie o TFTP e os serviços do CallManager em todos os Nós no conjunto que dirigem estes serviços.
4. Reinicie todos os Telefones IP de modo que possam obter o arquivo CTL do serviço TFTP CUCM.
5. Incorpore o comando do **ctl da mostra no** CLI a fim verificar o índice do arquivo CTL. No arquivo CTL, você pode ver que o certificado CCM+TFTP (server) do nó do editor CUCM está usado a fim assinar o arquivo CTL em vez do certificado dos eTokens do hardware USB. Uma diferença mais importante é neste caso que os Certificados de todos os eTokens do hardware USB estão removidos do arquivo CTL. Está aqui um exemplo de saída:

   ```
   admin:show ctl
   The checksum value of the CTL file:
   1d97d9089dd558a062cccfcb1dc4c57f(MD5)
   3b452f9ec9d6543df80e50f8b850cddc92fcf847(SHA1)

   Length of CTL file: 4947
   The CTL File was last modified on Fri Mar 06 21:56:07 CET 2015

   [...]
   ```

```
CTL Record #:1
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 System Administrator Security Token
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16  70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D
21 A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4
This etoken was used to sign the CTL file.

CTL Record #:2
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 CCM+TFTP
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D
21 A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4

[...]

The CTL file was verified successfully.
```

6. No lado do telefone IP, você pode verificar que depois que os Telefones IP foram reiniciados, transferiram a versão de arquivo actualizado CTL (a soma de verificação MD5 combina quando comparado à saída do CUCM):

Da solução de Tokenless aos eTokens do hardware

Esta seção descreve como migrar a Segurança do conjunto CUCM longe da solução nova de Tokenless e de volta ao uso de eTokens do hardware.

Quando a Segurança do conjunto CUCM está ajustada a modo misturado com o uso dos comandos CLI, e o arquivo CTL está assinado com o certificado CCM+TFTP (server) para o nó do editor CUCM, não há nenhum Certificados dos eTokens do hardware USB atuais no arquivo CTL. Por este motivo, quando você executa o cliente CTL a fim atualizar o arquivo CTL (movimento de volta ao uso de eTokens do hardware), este Mensagem de Erro aparece:

```
admin:show ctl
```

```
The checksum value of the CTL file:
1d97d9089dd558a062cccfcb1dc4c57f(MD5)
3b452f9ec9d6543df80e50f8b850cddc92fcf847(SHA1)

Length of CTL file: 4947
The CTL File was last modified on Fri Mar 06 21:56:07 CET 2015

[...]

CTL Record #:1
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 System Administrator Security Token
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16  70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D
21 A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4
This etoken was used to sign the CTL file.

CTL Record #:2
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 CCM+TFTP
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D
21 A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4

[...]

The CTL file was verified successfully.
```

Isto é particularmente importante nas encenações que incluem um downgrade (quando a versão é comutada para trás) do sistema a uma versão pre-10.x que não inclua os comandos do **ctl dos utils**. O arquivo precedente CTL é migrado (sem mudanças em seu índice) em processo de um refrescamento ou de um Linux à elevação de Linux (L2), e não contém os Certificados do eToken, como mencionado previamente. Está aqui um exemplo de saída:

```
admin:show ctl
The checksum value of the CTL file:
1d97d9089dd558a062cccfcb1dc4c57f(MD5)
3b452f9ec9d6543df80e50f8b850cddc92fcf847(SHA1)

Length of CTL file: 4947
```

```
The CTL File was last modified on Fri Mar 06 21:56:07 CET 2015

Parse CTL File
----------------

Version: 1.2
HeaderLength: 336 (BYTES)

BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
3 SIGNERID 2 149
4 SIGNERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
5 SERIALNUMBER 16 70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
6 CANAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
7 SIGNATUREINFO 2 15
8 DIGESTALGORTITHM 1
9 SIGNATUREALGOINFO 2 8
10 SIGNATUREALGORTITHM 1
11 SIGNATUREMODULUS 1
12 SIGNATURE 128
65 ba 26 b4 ba de 2b 13
b8 18 2 4a 2b 6c 2d 20
7d e7 2f bd 6d b3 84 c5
bf 5 f2 74 cb f2 59 bc
b5 c1 9f cd 4d 97 3a dd
6e 7c 75 19 a2 59 66 49
b7 64 e8 9a 25 7f 5a c8
56 bb ed 6f 96 95 c3 b3
72 7 91 10 6b f1 12 f4
d5 72 e 8f 30 21 fa 80
bc 5d f6 c5 fb 6a 82 ec
f1 6d 40 17 1b 7d 63 7b
52 f7 7a 39 67 e1 1d 45
b6 fe 82 0 62 e3 db 57
8c 31 2 56 66 c8 91 c8
d8 10 cb 5e c3 1f ef a
14 FILENAME 12
15 TIMESTAMP 4

CTL Record #:1
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 System Administrator Security Token
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16   70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D
21 A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4
This etoken was used to sign the CTL file.

CTL Record #:2
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
```

```
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 CCM+TFTP
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16   70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D
21 A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4

CTL Record #:3
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1138
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 60 CN=CAPF-e41e7d87;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 CAPF
5 ISSUERNAME 60 CN=CAPF-e41e7d87;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 74:4B:49:99:77:04:96:E7:99:E9:1E:81:D3:C8:10:9B
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 680 46 EE 5A 97 24 65 B0 17 7E 5F 7E 44 F7 6C 0A
F3 63 35 4F A7 (SHA1 Hash HEX)
10 IPADDRESS 4

CTL Record #:4
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1161
2 DNSNAME 17 cucm-1051-a-sub1
3 SUBJECTNAME 63 CN=cucm-1051-a-sub1;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 CCM+TFTP
5 ISSUERNAME 63 CN=cucm-1051-a-sub1;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 6B:EB:FD:CD:CD:8C:A2:77:CB:2F:D1:D1:83:A6:0E:72
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 696 21 7F 23 DE AF FF 04 85 76 72 70 BF B1 BA 44
DB 5E 90 ED 66 (SHA1 Hash HEX)
10 IPADDRESS 4

The CTL file was verified successfully.

admin:
```

Para esta encenação, termine estas etapas a fim atualizar firmemente os arquivos CTL sem a necessidade de usar o procedimento para eTokens perdidos, que termina acima na exclusão manual do arquivo CTL de todos os Telefones IP:

1. Obtenha o acesso administrativo ao nó CLI do editor CUCM.

2. Incorpore o **comando de tftp CTLFile.tlv da supressão do arquivo no** nó CLI do editor a fim suprimir do arquivo CTL:

```
admin:file delete tftp CTLFile.tlv
Delete the File CTLFile.tlv?
Enter "y" followed by return to continue: y
files: found = 1, deleted = 1
```

3. Abra o **cliente de autenticação de SafeNet na** máquina de Microsoft Windows que tem o cliente CTL instalado (é instalada automaticamente com cliente CTL):

4. No cliente de autenticação de SafeNet, navegue à *vista avançada*:

5. Introduza o primeiro hardware USB eToken.

6. Selecione o certificado sob o dobrador dos *certificados de usuário* e exporte-o para o dobrador no PC. Quando alertado para uma senha, use a senha padrão do **cisco123**:

7. Repita estas etapas para o segundo hardware USB eToken de modo que ambos os Certificados sejam exportados para o PC:

8. O log na administração unificada Cisco do operating system (OS) e navega ao **> gerenciamento de certificado da Segurança > ao certificado da transferência de arquivo pela rede**:

9. A página do certificado da transferência de arquivo pela rede publica-se então. Escolha a Telefone-SAST-**confiança da** finalidade do certificado deixam cair para baixo o menu e selecionam o certificado que você exportou do primeiro eToken:

10. Termine as etapas precedentes a fim transferir arquivos pela rede o certificado que você exportou do segundas eToken:

11. Execute o cliente CTL, forneça o endereço IP de Um ou Mais Servidores Cisco ICM NT/hostname do nó do editor CUCM, e incorpore as credenciais do administrador CCM:

12. Desde que o conjunto reage de modo misturado já, mas nenhum arquivo CTL existe no nó

do editor, este mensagem de advertência aparece (**APROVAÇÃO do** clique a fim a ignorar):

```
admin:file delete tftp CTLFile.tlv
Delete the File CTLFile.tlv?
Enter "y" followed by return to continue: y
files: found = 1, deleted = 1
```

13. Do cliente CTL, clique o botão de rádio do **arquivo da atualização CTL**, e clique-o então **em seguida**:

14. Introduza o primeiro token de segurança e clique a **APROVAÇÃO**:

15. Depois que os detalhes do token de segurança são indicados, o clique **adiciona**:

16. Uma vez o índice do arquivo CTL aparece, clique **adiciona tokens** a fim adicionar o segundo USB eToken:

17. Depois que os detalhes do token de segurança aparecem, o clique **adiciona**:

18. Depois que o índice do arquivo CTL aparece, clique o **revestimento**. Quando alertado para uma senha, entre no **cisco123**:

19. Quando a lista de server CUCM em que o arquivo CTL existe aparecer, clique **feito**:

20. Reinicie o TFTP e os serviços do CallManager em todos os Nós no conjunto que dirigem estes serviços.

21. Reinicie todos os Telefones IP de modo que possam obter a nova versão do arquivo CTL do serviço TFTP CUCM.

22. A fim verificar o índice do arquivo CTL, incorpore o comando do **ctl da mostra no** CLI. No arquivo CTL você pode ver os Certificados de ambos os eTokens USB (um deles é usado a fim assinar o arquivo CTL). Está aqui um exemplo de saída:

```
admin:show ctl
```

```
The checksum value of the CTL file:
2e7a6113eadbdae67ffa918d81376902(MD5)
d0f3511f10eef775cc91cce3fa6840c2640f11b8(SHA1)

Length of CTL file: 5728
The CTL File was last modified on Fri Mar 06 22:53:33 CET 2015

[...]

CTL Record #:1
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1186
2 DNSNAME 1
3 SUBJECTNAME 56 cn="SAST-ADN0054f509 ";ou=IPCBU;o="Cisco Systems
4 FUNCTION 2  System Administrator Security Token
5 ISSUERNAME 42 cn=Cisco Manufacturing CA;o=Cisco Systems
6 SERIALNUMBER 10  3C:F9:27:00:00:00:AF:A2:DA:45
7 PUBLICKEY 140
9 CERTIFICATE 902 19 8F 07 C4 99 20 13 51 C5 AE BF 95 03 93 9F F2
CC 6D 93 90 (SHA1 Hash HEX)
10 IPADDRESS 4
This etoken was not used to sign the CTL file.

[...]

CTL Record #:5
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1186
2 DNSNAME 1
3 SUBJECTNAME 56 cn="SAST-ADN008580ef ";ou=IPCBU;o="Cisco Systems
4 FUNCTION 2  System Administrator Security Token
5 ISSUERNAME 42 cn=Cisco Manufacturing CA;o=Cisco Systems
6 SERIALNUMBER 10 83:E9:08:00:00:00:55:45:AF:31
7 PUBLICKEY 140
9 CERTIFICATE 902 85 CD 5D AD EA FC 34 B8 3E 2F F2 CB 9C 76 B0 93
3E 8B 3A 4F (SHA1 Hash HEX)
10 IPADDRESS 4
This etoken was used to sign the CTL file.

The CTL file was verified successfully.
```

23. No lado do telefone IP, você pode verificar que depois que os Telefones IP foram reiniciados, transferiram a versão de arquivo actualizado CTL (a soma de verificação MD5 combina quando comparado à saída do CUCM):

Esta mudança é possível porque você previamente exportou e transferiu arquivos pela rede os Certificados do eToken à loja da confiança do certificado CUCM, e os Telefones IP podem verificar este certificado desconhecido que foi usado a fim assinar o arquivo CTL contra o serviço da verificação da confiança (TV) essas corridas no CUCM. Este snippit do log ilustra como o telefone IP contacta o CUCM TV com um pedido verificar que o desconhecido eToken o certificado, que é transferido arquivos pela rede como a Telefone-SAST-confiança e confiado:

```
//In the Phone Console Logs we can see a request sent to TVS server to verify unknown
certificate
8074: NOT 23:00:22.335499 SECD: setupSocketToTvsProxy: Connected to TVS proxy server
8075: NOT 23:00:22.336918 SECD: tvsReqFlushTvsCertCache: Sent Request to TVS proxy,
len: 3708
```

```
//In the TVS logs on CUCM we can see the request coming from an IP Phone which is being
successfully verified
23:00:22.052 | debug tvsHandleQueryCertReq
23:00:22.052 | debug tvsHandleQueryCertReq : Subject Name is: cn="SAST-ADN008580ef
";ou=IPCBU;o="Cisco Systems
23:00:22.052 | debug tvsHandleQueryCertReq : Issuer Name is: cn=Cisco Manufacturing
CA;o=Cisco Systems
23:00:22.052 | debug tvsHandleQueryCertReq :subjectName and issuerName matches for
eToken certificate
23:00:22.052 | debug tvsHandleQueryCertReq : SAST Issuer Name is: cn=Cisco
Manufacturing CA;o=Cisco Systems
23:00:22.052 | debug tvsHandleQueryCertReq : This is SAST eToken cert
23:00:22.052 | debug tvsHandleQueryCertReq : Serial Number is: 83E9080000005545AF31
23:00:22.052 | debug CertificateDBCache::getCertificateInformation - Looking up the
certificate cache using Unique MAP ID : 83E9080000005545AF31cn=Cisco Manufacturing
CA;o=Cisco Systems
23:00:22.052 | debug ERROR:CertificateDBCache::getCertificateInformation - Cannot find
the certificate in the cache
23:00:22.052 | debug CertificateCTLCache::getCertificateInformation - Looking up the
certificate cache using Unique MAP ID : 83E9080000005545AF31cn=Cisco Manufacturing
CA;o=Cisco Systems, len : 61
23:00:22.052 | debug CertificateCTLCache::getCertificateInformation - Found entry
{rolecount : 1}
23:00:22.052 | debug CertificateCTLCache::getCertificateInformation - {role : 0}
23:00:22.052 | debug convertX509ToDER -x509cert : 0xa3ea6f8
23:00:22.053 | debug tvsHandleQueryCertReq: Timer started from tvsHandleNewPhConnection

//In the Phone Console Logs we can see reply from TVS server to trust the new certificate
(eToken Certificate which was used to sign the CTL file)
8089: NOT 23:00:22.601218 SECD: clpTvsInit: Client message received on TVS proxy socket
8090: NOT 23:00:22.602785 SECD: processTvsClntReq: Success reading the client TVS
request, len : 3708
8091: NOT 23:00:22.603901 SECD: processTvsClntReq: TVS Certificate cache flush
request received
8092: NOT 23:00:22.605720 SECD: tvsFlushCertCache: Completed TVS Certificate cache
flush request
```

Regeneração do certificado para a solução de Tokenless CTL

Esta seção descreve como regenerar um Security Certificate do conjunto CUCM quando a solução de Tokenless CTL é usada.

Em processo da manutenção CUCM, às vezes o certificado do CallManager do nó do editor CUCM muda. As encenações em que esta pode acontecer incluem a mudança do hostname, a mudança do domínio, ou simplesmente uma regeneração do certificado (devendo fechar a data de expiração do certificado).

Depois que o arquivo CTL é atualizado, está assinado com um certificado diferente do que aqueles que existem no arquivo CTL que é instalado nos Telefones IP. Normalmente, este arquivo novo CTL não é aceitado; contudo, depois que o telefone IP encontra o certificado desconhecido que está usado a fim assinar o arquivo CTL, contacta o serviço TV no CUCM.

> **Note**: A lista de servidor TV está no arquivo de configuração de telefone IP e é traçada nos server CUCM do **pool de dispositivos > do grupo do CallManager do** telefone IP.

Em cima da verificação bem-sucedida contra o server TV, o telefone IP atualiza seu arquivo CTL

com a nova versão. Estes eventos ocorrem em tal encenação:

1. O arquivo CTL existe no CUCM e no telefone IP. O certificado CCM+TFT (server) para o nó do editor CUCM é usado a fim assinar o arquivo CTL:

```
admin:show ctl
The checksum value of the CTL file:
7b7c10c4a7fa6de651d9b694b74db25f(MD5)
819841c6e767a59ecf2f87649064d8e073b0fe87(SHA1)

Length of CTL file: 4947
The CTL File was last modified on Mon Mar 09 16:59:43 CET 2015

[...]

CTL Record #:1
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 System Administrator Security Token
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D
21 A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4
This etoken was used to sign the CTL file.

CTL Record #:2
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1156
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 CCM+TFTP
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB
7 PUBLICKEY 140
8 SIGNATURE 128
9 CERTIFICATE 694 E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D
21 A5 A3 8C 9C (SHA1 Hash HEX)
10 IPADDRESS 4

[...]

The CTL file was verified successfully.
```

2. O arquivo CallManager.pem (certificado CCM+TFTP) é regenerado, e você pode considerar que o número de série do certificado muda:

3. O comando de **CTLFile da atualização do ctl dos utils** é incorporado no CLI a fim atualizar o arquivo CTL:

```
admin:utils ctl update CTLFile
This operation will update the CTLFile. Do you want to continue? (y/n):y

Updating CTL file
CTL file Updated
Please Restart the TFTP and Cisco CallManager services on all nodes in
the cluster that run these services
admin:
```

4. O serviço TV atualiza seu esconderijo do certificado com os detalhes novos do arquivo CTL:

```
17:10:35.825 | debug CertificateCache::localCTLCacheMonitor - CTLFile.tlv has been
modified. Recaching CTL Certificate Cache
17:10:35.826 | debug updateLocalCTLCache : Refreshing the local CTL certificate cache
17:10:35.827 | debug tvs_sql_get_all_CTL_certificate - Unique Key used for Caching ::
6B1D357B6841740B078FEE4A1813D5D6CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL, length : 93
17:10:35.827 | debug tvs_sql_get_all_CTL_certificate - Unique Key used for Caching ::
6B1D357B6841740B078FEE4A1813D5D6CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL, length : 93
17:10:35.827 | debug tvs_sql_get_all_CTL_certificate - Unique Key used for Caching ::
744B5199770516E799E91E81D3C8109BCN=CAPF-e41e7d87;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL, length : 91
17:10:35.827 | debug tvs_sql_get_all_CTL_certificate - Unique Key used for Caching ::
6BEBFDCDCD8CA277CB2FD1D183A60E72CN=cucm-1051-a-sub1;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL, length : 94
```

5. Quando você vê o conteúdo de arquivo CTL, você pode ver que o arquivo está assinado com o certificado de servidor do CallManager novo para o nó do editor:

```
admin:show ctl
The checksum value of the CTL file:
ebc649598280a4477bb3e453345c8c9d(MD5)
ef5c006b6182cad66197fac6e6530f15d009319d(SHA1)

Length of CTL file: 6113
The CTL File was last modified on Mon Mar 09 17:07:52 CET 2015

[..]

CTL Record #:1
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1675
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 System Administrator Security Token
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 6B:1D:35:7B:68:41:74:0B:07:8F:EE:4A:18:13:D5:D6
7 PUBLICKEY 270
8 SIGNATURE 256
9 CERTIFICATE 955 5C AF 7D 23 FE 82 DB 87 2B 6F 4D B7 F0 9D D5
86 EE E0 8B FC (SHA1 Hash HEX)
```

```
10 IPADDRESS 4

This etoken was used to sign the CTL file.

CTL Record #:2
----
BYTEPOS TAG LENGTH VALUE
------- --- ------ -----
1 RECORDLENGTH 2 1675
2 DNSNAME 16 cucm-1051-a-pub
3 SUBJECTNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
4 FUNCTION 2 CCM+TFTP
5 ISSUERNAME 62 CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
ST=Malopolska;C=PL
6 SERIALNUMBER 16 6B:1D:35:7B:68:41:74:0B:07:8F:EE:4A:18:13:D5:D6
7 PUBLICKEY 270
8 SIGNATURE 256
9 CERTIFICATE 955 5C AF 7D 23 FE 82 DB 87 2B 6F 4D B7 F0 9D D5
86 EE E0 8B FC (SHA1 Hash HEX)
10 IPADDRESS 4

[...]

The CTL file was verified successfully.
```

6. Da página unificada da utilidade, o TFTP e os serviços do CallManager da Cisco são reiniciados em todos os Nós no conjunto que dirigem estes serviços.

7. Os Telefones IP são reiniciados, e contactam o server TV a fim verificar o certificado desconhecido que é usado agora a fim assinar a nova versão do arquivo CTL:

```
// In the Phone Console Logs we can see a request sent to TVS server to verify
unknown certificate
2782: NOT 17:21:51.794615 SECD: setupSocketToTvsProxy: Connected to TVS proxy server
2783: NOT 17:21:51.796021 SECD: tvsReqFlushTvsCertCache: Sent Request to TVS
proxy, len: 3708

// In the TVS logs on CUCM we can see the request coming from an IP Phone which is
being successfully verified
17:21:51.831 | debug tvsHandleQueryCertReq
17:21:51.832 | debug tvsHandleQueryCertReq : Subject Name is: CN=cucm-1051-a-pub;
OU=TAC;O=Cisco;L=Krakow;ST=Malopolska
17:21:51.832 | debug tvsHandleQueryCertReq : Issuer Name is: CN=cucm-1051-a-pub;
OU=TAC;O=Cisco;L=Krakow;ST=Malopolska;
17:21:51.832 | debug tvsHandleQueryCertReq : Serial Number is:
6B1D357B6841740B078FEE4A1813D5D6
17:21:51.832 | debug CertificateDBCache::getCertificateInformation - Looking up the
certificate cache using Unique MAPco;L=Krakow;ST=Malopolska;C=PL
17:21:51.832 | debug CertificateDBCache::getCertificateInformation - Found entry
{rolecount : 2}
17:21:51.832 | debug CertificateDBCache::getCertificateInformation - {role : 0}
17:21:51.832 | debug CertificateDBCache::getCertificateInformation - {role : 2}
17:21:51.832 | debug convertX509ToDER -x509cert : 0xf6099df8
17:21:51.832 | debug tvsHandleQueryCertReq: Timer started from
tvsHandleNewPhConnection

// In the Phone Console Logs we can see reply from TVS server to trust the new
certificate (new CCM Server Certificate which was used to sign the CTL file)
2797: NOT 17:21:52.057442 SECD: clpTvsInit: Client message received on TVS
proxy socket
2798: NOT 17:21:52.058874 SECD: processTvsClntReq: Success reading the client TVS
request, len : 3708
```

```
2799: NOT 17:21:52.059987 SECD: processTvsClntReq: TVS Certificate cache flush
request received
2800: NOT 17:21:52.062873 SECD: tvsFlushCertCache: Completed TVS Certificate
cache flush request
```

8. Finalmente, nos Telefones IP, você pode verificar que o arquivo CTL está atualizado com a nova versão e que a soma de verificação MD5 do arquivo novo CTL combina com a aquela do CUCM: