Modo mixto de CUCM con CTL sin tokens

Contenido

Introducción

En este documento, se describe la diferencia entre la seguridad de Cisco Unified Communications Manager (CUCM) con y sin el uso de eTokens USB de hardware. Este documento también describe las situaciones básicas de implementación que implican la Lista de confianza de certificados sin tokens (CTL) y el proceso que se utiliza para garantizar que el sistema funcione correctamente después de los cambios.

Prerequisites

Requirements

Cisco recomienda tener conocimientos de la versión 10.0(1) de CUCM o una posterior. Además, asegúrese de lo siguiente:

- El servidor de licencias para la versión 11.5.1SU3 de CUCM y versiones posteriores debe ser Cisco Prime License Manager (PLM) 11.5.1SU2 o superior. Esto se debe a que la versión 11.5.1SU3 de CUCM requiere la Licencia de cifrado para habilitar el modo mixto y PLM no admite la Licencia de cifrado hasta la versión 11.5.1SU2. Para obtener más información, consulte las Notas de la versión 11.5(1)SU2 de Cisco Prime License Manager.

- Tiene acceso administrativo a la Interfaz de línea de comandos (CLI) del nodo de editor de CUCM.

- Tiene acceso a los eTokens USB de hardware y el complemento del cliente CTL está instalado en su PC para situaciones que requieran que migre nuevamente al uso de eTokens de hardware. Para mayor claridad, este requisito solo se aplica si en algún momento se presenta una situación en la que se necesitan los eTokens USB. Hay muy pocas probabilidades de que la mayoría de las personas necesiten eTokens USB.

- Hay conectividad total entre todos los nodos de CUCM en el clúster. Esto es muy importante

porque el archivo CTL se copia a todos los nodos del clúster mediante el protocolo de transferencia de archivos (SFTP) de SSH.

- La Replicación de la base de datos (DB) en el clúster funciona correctamente y los servidores replican los datos en tiempo real.

- Los dispositivos en la implementación admiten la Seguridad de manera predeterminada (TVS). Puede utilizar la *Lista de funciones de teléfono de Unified CM de la página web de Cisco Unified Reporting (https://<CUCM IP or FQDN>cucreports/) para determinar los dispositivos que admiten la Seguridad de manera predeterminada.*

  Nota: En la actualidad, Cisco Jabber y muchos teléfonos IP de Cisco TelePresence o de la serie 7940/7960 de Cisco no admiten la Seguridad de manera predeterminada. Si implementa CTL sin tokens con dispositivos que no admiten la Seguridad de manera predeterminada, cualquier actualización en el sistema que cambie el certificado CallManager en el editor evitará la funcionalidad normal de dichos dispositivos hasta que se elimine manualmente la CTL. Los dispositivos que admiten la Seguridad de manera predeterminada, como los teléfonos 7945 y 7965 o más nuevos, pueden instalar archivos CTL cuando se actualiza el certificado de CallManager en el editor, ya que pueden utilizar el Servicio de verificación de confianza (TVS).

Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Versión 10.5.1.10000-7 de CUCM (clúster de dos nodos)

- Teléfonos IP de la serie 7975 de Cisco registrados mediante el Protocolo de control de cliente Skinny (SCCP) con versión de firmware SCCP75.9-3-1SR4-1S

- Dos tokens de seguridad de Cisco que se utilizan para establecer el clúster en modo mixto con el uso del software de cliente CTL

Antecedentes

La CTL sin tokens es una nueva función en las versiones 10.0(1) de CUCM y posteriores, la cual permite el cifrado de la señalización de llamadas y los medios para los teléfonos IP sin necesidad de utilizar eTokens USB de hardware y el complemento de Cliente CTL, que era el requisito de las versiones anteriores de CUCM.

Cuando el clúster se coloca en modo mixto con el uso del comando CLI, el archivo CTL se firma con el certificado CCM+TFTP (servidor) del nodo de editor y no hay certificados eToken presentes en el archivo CTL.

Nota: Cuando vuelve a generar el certificado CallManager (CCM+TFTP) en el editor, cambia el firmante del archivo. Los teléfonos y dispositivos que no admiten la Seguridad de manera predeterminada, no aceptarán el nuevo archivo CTL, a menos que los archivos CTL se eliminen manualmente de cada dispositivo. Para más información, consulte el último

requisito que aparece en la sección Requisitos de este documento.

De modo no seguro a modo mixto (CTL sin tokens)

En esta sección, se describe el proceso que se utiliza para mover la seguridad del clúster de CUCM al modo mixto mediante la CLI.

Antes de esta situación, el CUCM se encontraba en modo no seguro, lo que significa que no había ningún archivo CTL presente en ninguno de los nodos y que los teléfonos IP registrados solo tenían un archivo de Lista de confianza de identidad (ITL) instalado, como se muestra en estos resultados:

```
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:
```

Nota: Si hay un archivo CTL que se encuentra en el servidor mientras el clúster no está en modo mixto, esto significa que el clúster estuvo una vez en modo mixto y luego se volvió a pasar al modo no mixto y el archivo CTL no se eliminó del clúster.

El archivo de comandos delete activelog cm/tftpdata/CTLFile.tlv elimina el archivo CTL de los nodos en el clúster de CUCM; sin embargo, se debe ingresar el comando en cada nodo. Para que quede claro, utilice este comando solo si sus servidores tienen un archivo CTL y el clúster no está en modo mixto.

Una manera fácil de confirmar si un clúster está en modo mixto es utilizar el comando **run sql select paramname,paramvalue from processconfig where paramname='ClusterSecurityMode'.** Si el valor del parámetro es 0, el clúster no está en modo mixto.

```
run sql select paramname,paramvalue from processconfig where paramname='ClusterSecurityMode'
paramname           paramvalue
=================== ==========
ClusterSecurityMode 0
```

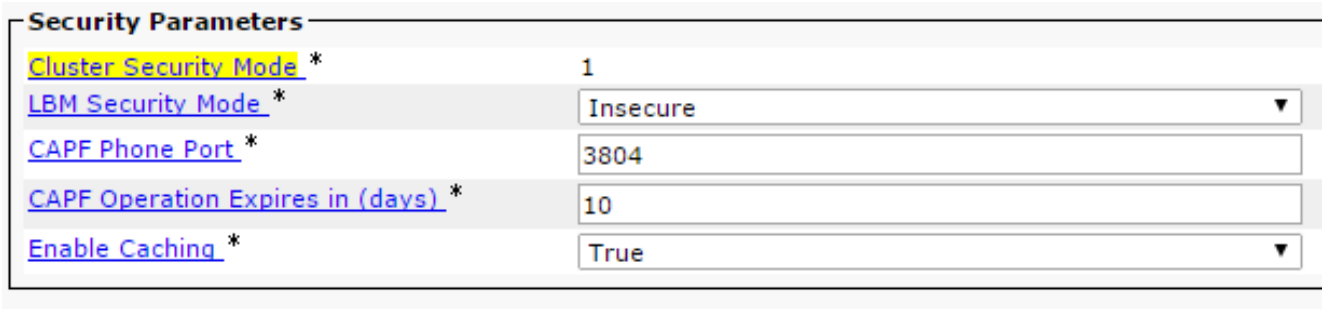
Para pasar la seguridad del clúster de CUCM a modo mixto con el uso de la nueva función de CTL sin tokens, siga estos pasos:

1. Obtenga acceso administrativo a la CLI del nodo de editor de CUCM.

2. Introduzca el comando **utils ctl set-cluster mixed-mod en la 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. Vaya a la **Página de administrador de CUCM > Sistema > Parámetros de la empresa** y verifique si el clúster se configuró en modo mixto (un valor de **1 indica modo mixto):**



4. Reinicie el TFTP y los servicios de Cisco CallManager en todos los nodos del clúster que ejecutan estos servicios.

5. Reinicie todos los teléfonos IP para que puedan obtener el archivo CTL del servicio TFTP de CUCM.

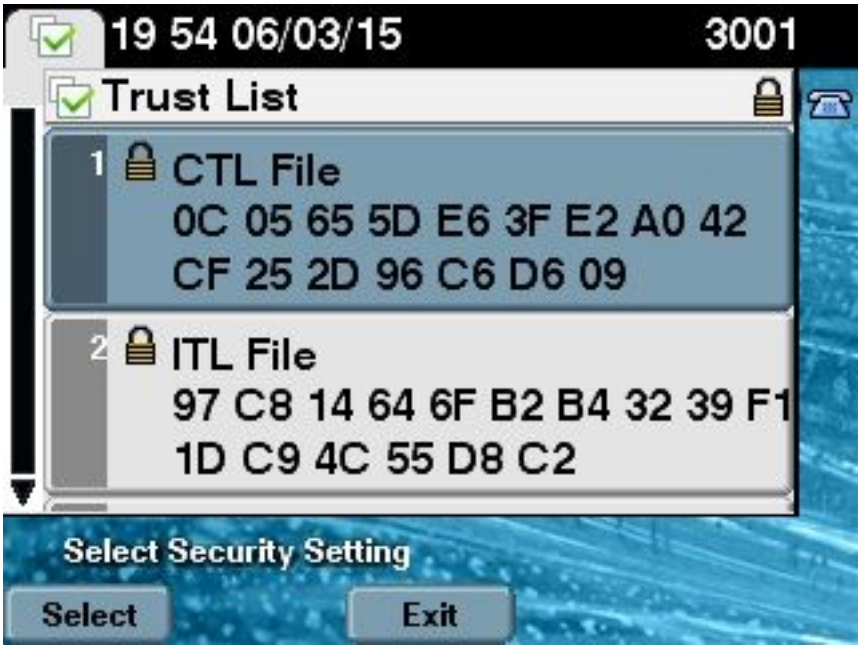
6. Para verificar el contenido del archivo CTL, introduzca el comando **show CTL en la CLI.** En

el archivo CTL, puede ver que el certificado CCM+TFTP (servidor) para el nodo de editor de CUCM se utiliza para firmar el archivo CTL (este archivo es el mismo en todos los servidores del clúster). Éste es un ejemplo de salida:

```
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. En el lado del teléfono IP, puede verificar que, después de que se reinicia el servicio, descarga el archivo CTL, que ahora está presente en el servidor TFTP (las coincidencias de checksum MD5 comparadas con el resultado del CUCM):

Nota: Cuando verifica la suma de comprobación en el teléfono, ve **MD5** o **SHA1**, según el tipo de teléfono.



De eTokens de hardware a una solución sin tokens

En esta sección, se describe cómo migrar la seguridad del clúster de CUCM de eTokens de hardware al uso de la nueva solución sin tokens.

En algunas situaciones, el modo mixto ya está configurado en el CUCM con el uso del cliente de CTL, y los teléfonos IP utilizan archivos CTL que contienen los certificados de los eTokens USB de hardware. Con esta situación, el archivo CTL está firmado por un certificado de uno de los eTokens USB y está instalado en los teléfonos IP. El siguiente es un ejemplo:

```
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.
```

Complete estos pasos para pasar la seguridad del clúster de CUCM al uso de las CTL sin tokens:

1. Obtenga acceso administrativo a la CLI del nodo de editor de CUCM.

2. Introduzca el comando **utils ctl update CTLFile de la CLI:**

```
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 los servicios de TFTP y CallManager en todos los nodos del clúster que ejecutan estos servicios.

4. Reinicie todos los teléfonos IP para que puedan obtener el archivo CTL del servicio TFTP de CUCM.

5. Para verificar el contenido del archivo CTL, introduzca el comando **show ctl** en la CLI. En el archivo CTL, puede ver que el certificado CCM+TFTP (servidor) del nodo de editor de CUCM se utiliza para firmar el archivo CTL en lugar del certificado de los eTokens USB de hardware. Otra diferencia importante en este caso es que los certificados de todos los eTokens USB de hardware se eliminan del archivo CTL. Éste es un ejemplo de salida:

```
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.
```

Nota: En el resultado anterior, si el certificado de CCM+TFTP (servidor) del editor de CUCM no es firmante, vuelva al modo de seguridad de clúster basado en el eToken de hardware y repita los cambios nuevamente para la solución sin tokens.

6. En el lado del teléfono IP, puede verificar que, después de que se reiniciaron los teléfonos IP, descargaron la versión actualizada del archivo CTL (las coincidencias de checksum MD5 comparadas con el resultado del CUCM):

De una solución sin tokens a eTokens de hardware

En esta sección, se describe cómo migrar la seguridad del clúster de CUCM de la nueva solución sin tokens a usar nuevamente los eTokens de hardware.

Cuando la seguridad del clúster de CUCM se coloca en modo mixto con el uso de los comandos de la CLI y el archivo CTL se firma con el certificado CCM+TFTP (servidor) del nodo de editor de CUCM, no hay certificados de los eTokens USB presentes en el archivo CTL. Por este motivo, cuando ejecuta el cliente CTL para actualizar el archivo CTL (volver a usar eTokens de hardware), aparece este mensaje de error:

```
The Security Token you have inserted does not exist in the CTL File
Please remove any Security Tokens already inserted and insert another
Security Token. Click Ok when done.
```

Esto es especialmente importante en situaciones que incluyen una actualización del sistema a una versión anterior a 10.x que no incluya los comandos **utils ctl** . El archivo CTL anterior se migra (sin cambios en su contenido) en el proceso de una actualización, por ej., de Linux a Linux (L2), y no contiene los certificados eToken, como se mencionó anteriormente. Éste es un ejemplo de salida:

```
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:
```

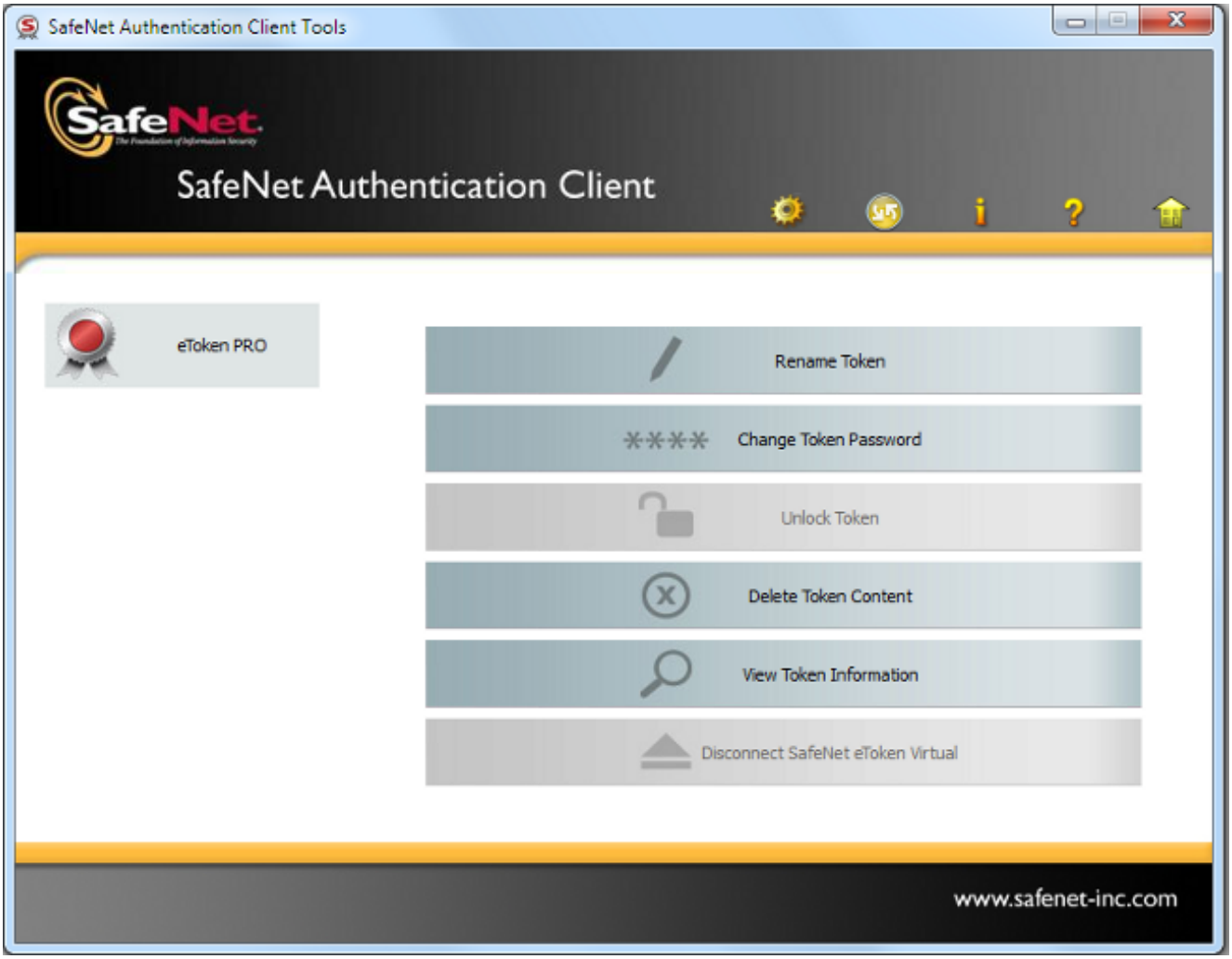
Para esta situación, siga estos pasos para actualizar de manera segura los archivos de CTL, sin la necesidad de utilizar el procedimiento para eTokens perdidos, que termina en la eliminación manual del archivo CTL de todos los teléfonos IP:

1. Obtenga acceso administrativo a la CLI del nodo de editor de CUCM.

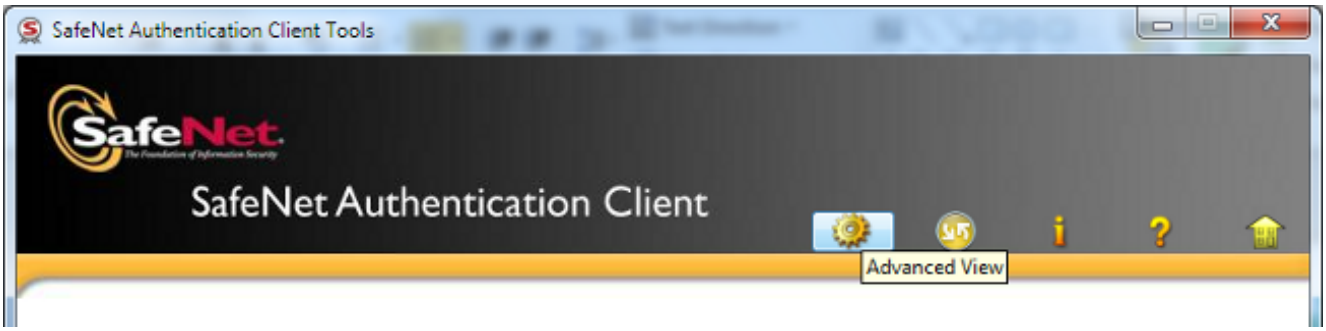
2. Introduzca el **comando file delete tftp CTLFile.tlv en la CLI del nodo de editor para eliminar el archivo CTL:**

   ```
   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 el **Cliente de autenticación de SafeNet en la máquina de Windows Microsoft que tiene el cliente CTL instalado (se instala automáticamente con el cliente CTL):**

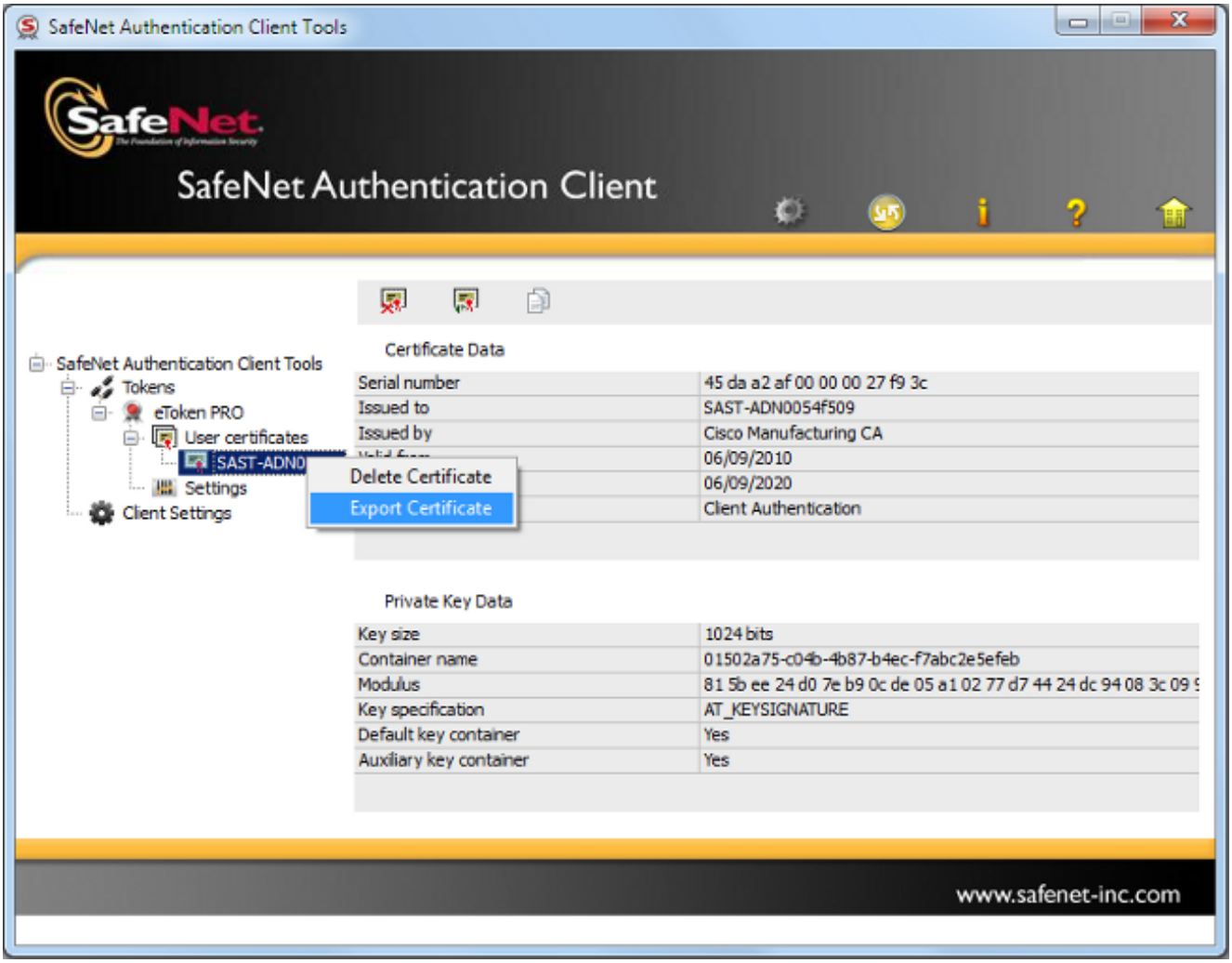


4. En el cliente de autenticación de SafeNet, *navegue hasta*la vista avanzada:

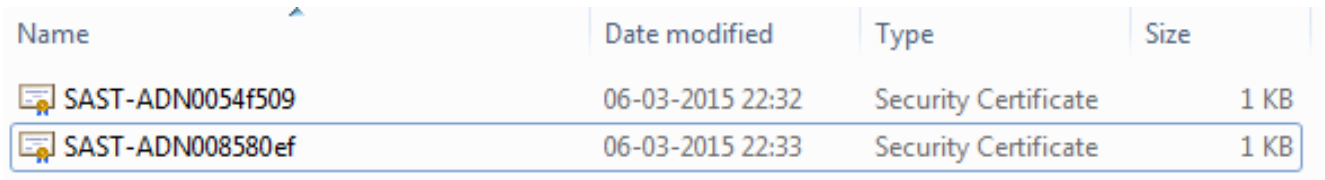


5. Inserte el primer eToken USB de hardware.

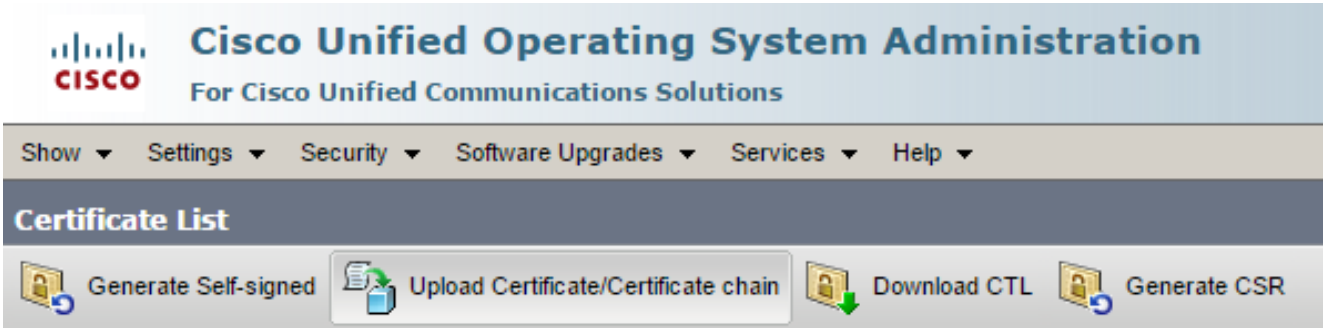
6. Seleccione el certificado en la carpeta *Certificados de usuario y expórtelo a la carpeta de la PC.* Cuando se le solicite una contraseña, utilice la contraseña predeterminada, **Cisco123**:



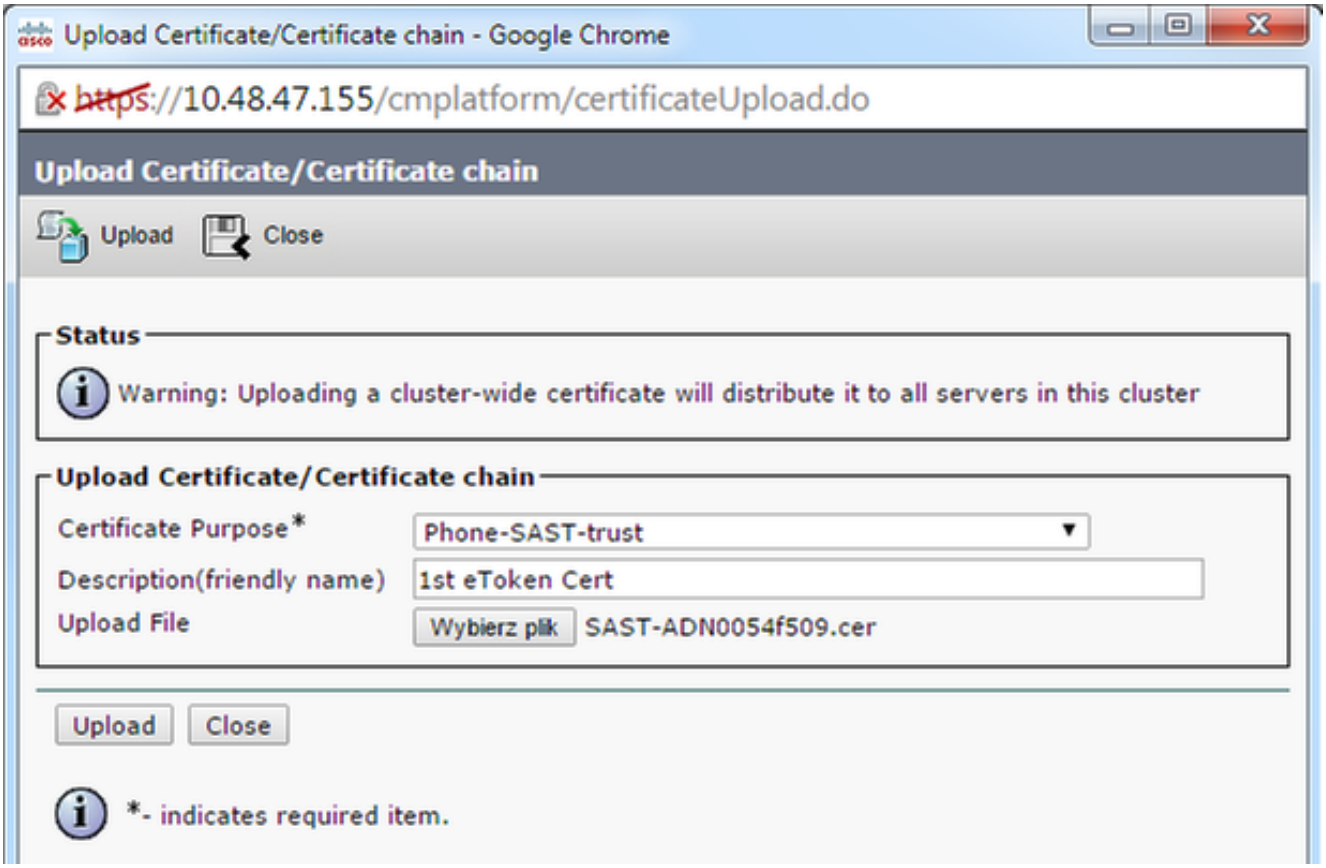
7. Repita estos pasos para el segundo eToken USB de hardware, de modo que ambos certificados se exporten a la PC:



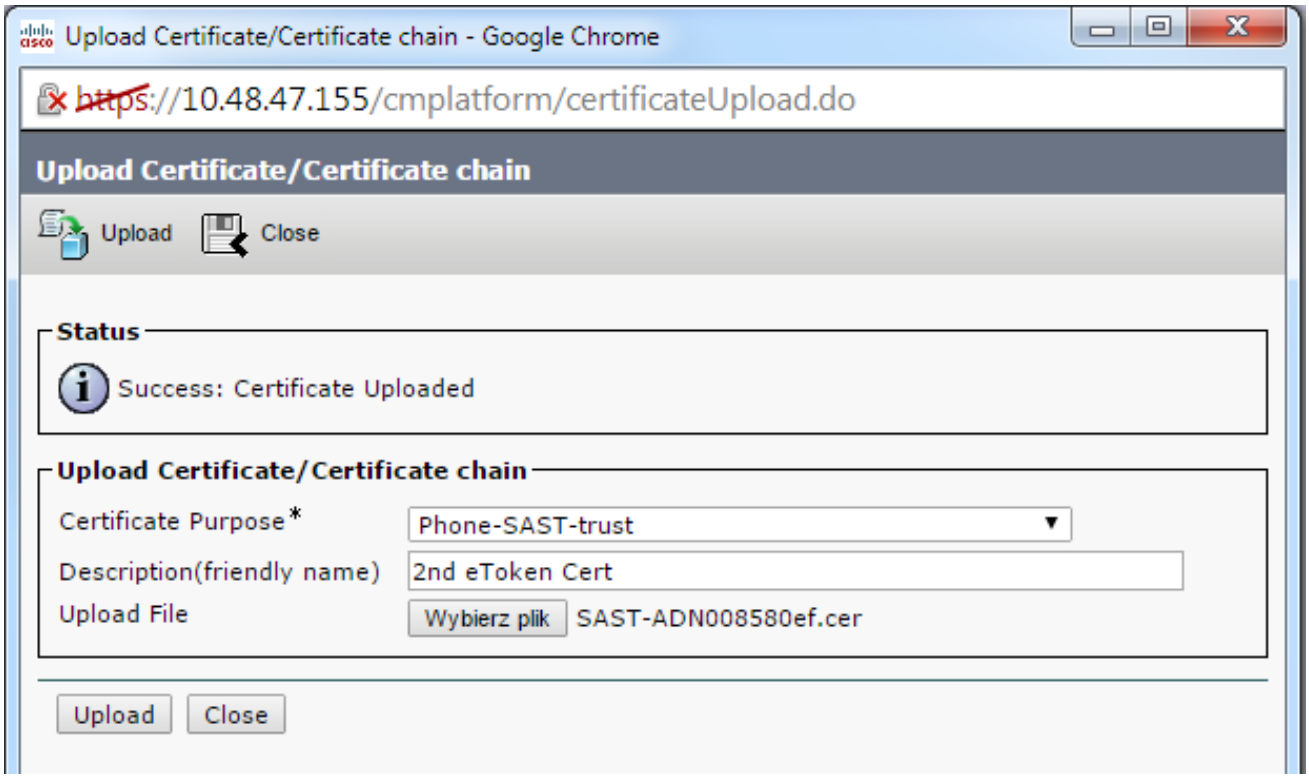
8. Inicie sesión en la Administración de Cisco Unified Operating System (SO) y navegue a **Seguridad > Administración de certificados > Cargar certificado**:



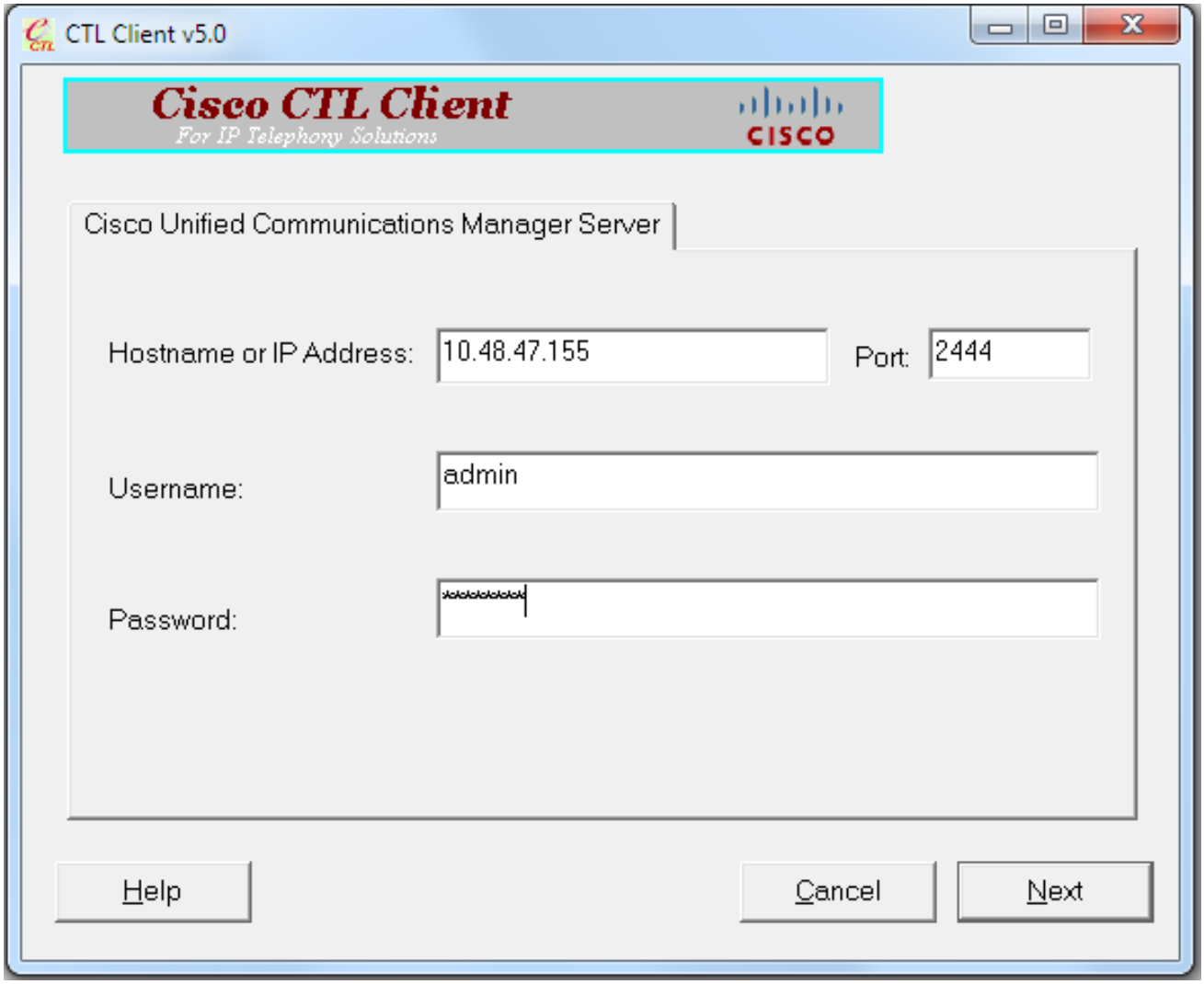
9. Aparecerá la página Cargar certificado. Elija **Phone-SAST-trust en el menú desplegable Propósito del certificado y seleccione el certificado que exportó del primer eToken:**



10. Complete los pasos anteriores para cargar el certificado exportado desde el segundo eToken:



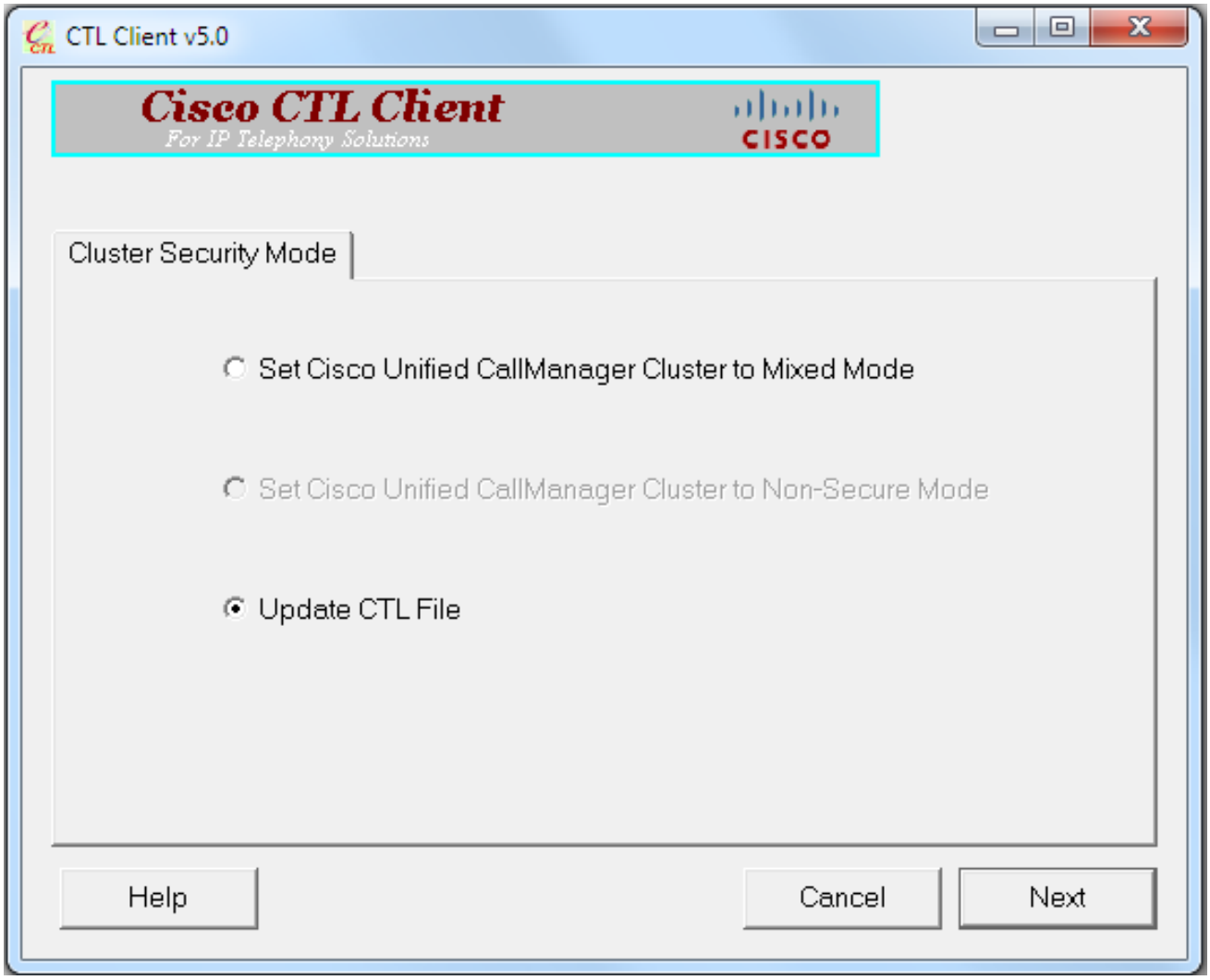
11. Ejecute el Cliente CTL, proporcione la dirección IP/nombre de host del nodo de editor de CUCM e introduzca las credenciales de administrador de CCM:



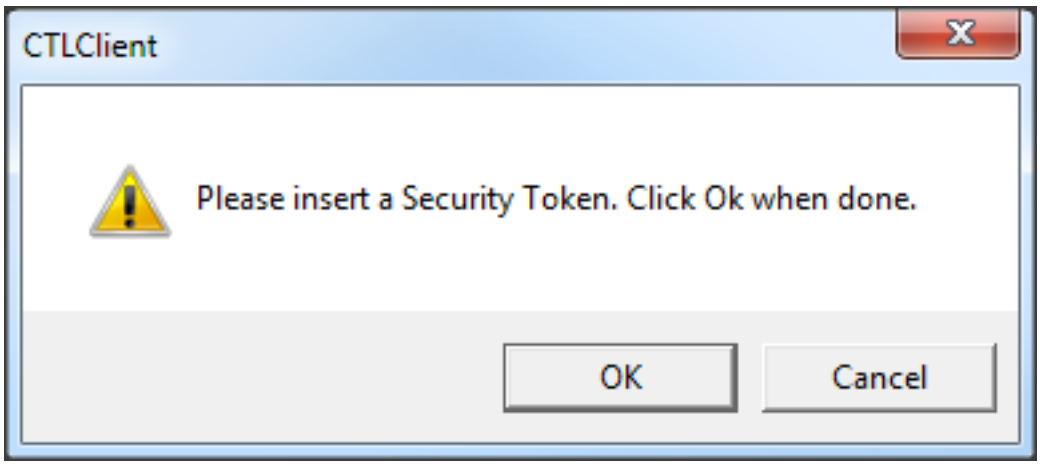
12. Dado que el clúster ya está en modo mixto, pero no existe ningún archivo CTL en el nodo de editor, aparece este mensaje de advertencia (haga clic en **Aceptar para ignorarlo):**

```
No CTL File exists on the server but the Call Manager Cluster Security Mode
is in Secure Mode.
For the system to function, you must create the CTL File and set Call Manager
Cluster the Secure Mode.
```

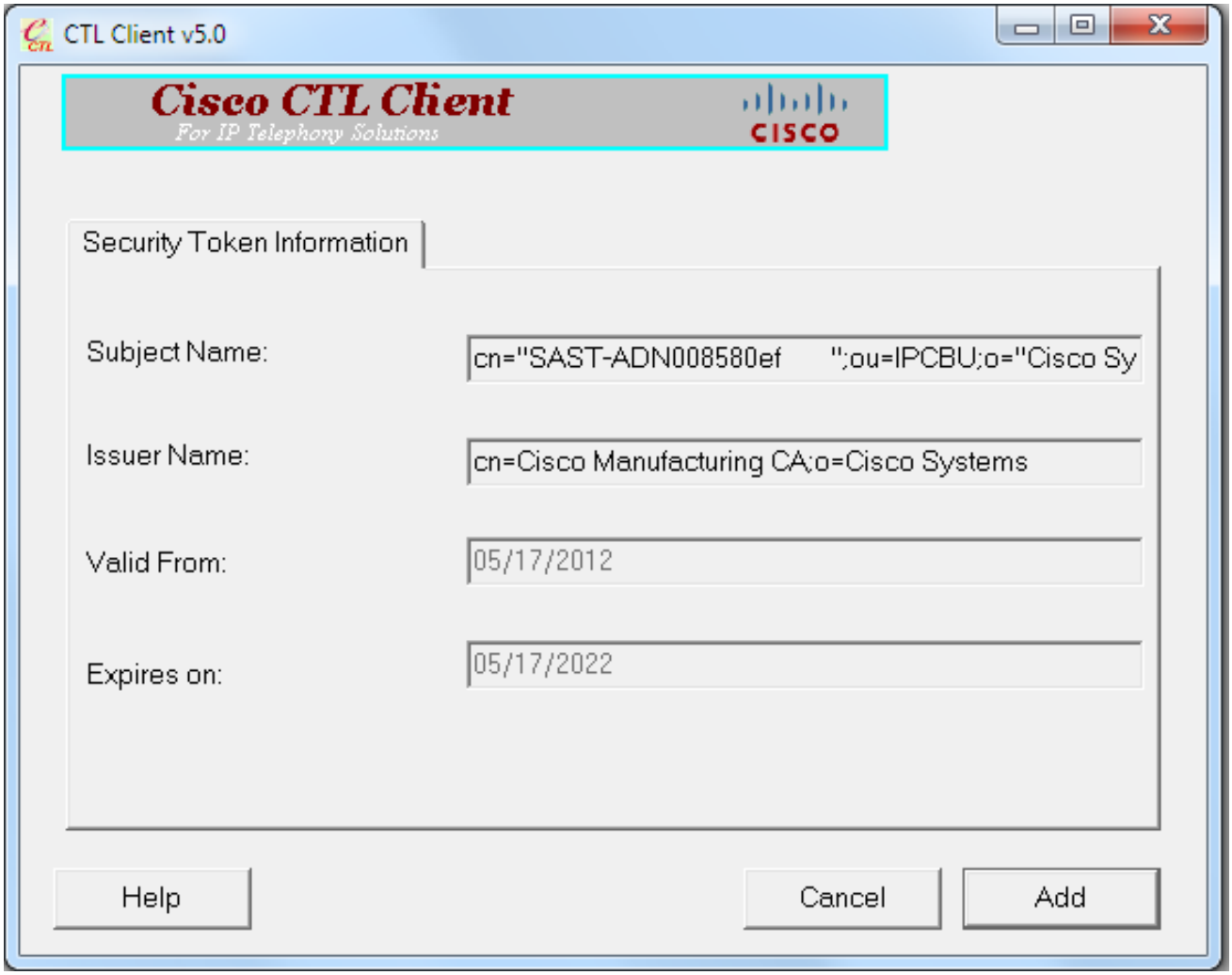
13. Desde el cliente CTL, haga clic en el botón de opción **Actualizar archivo CTL y, a continuación, haga clic en Siguiente**:



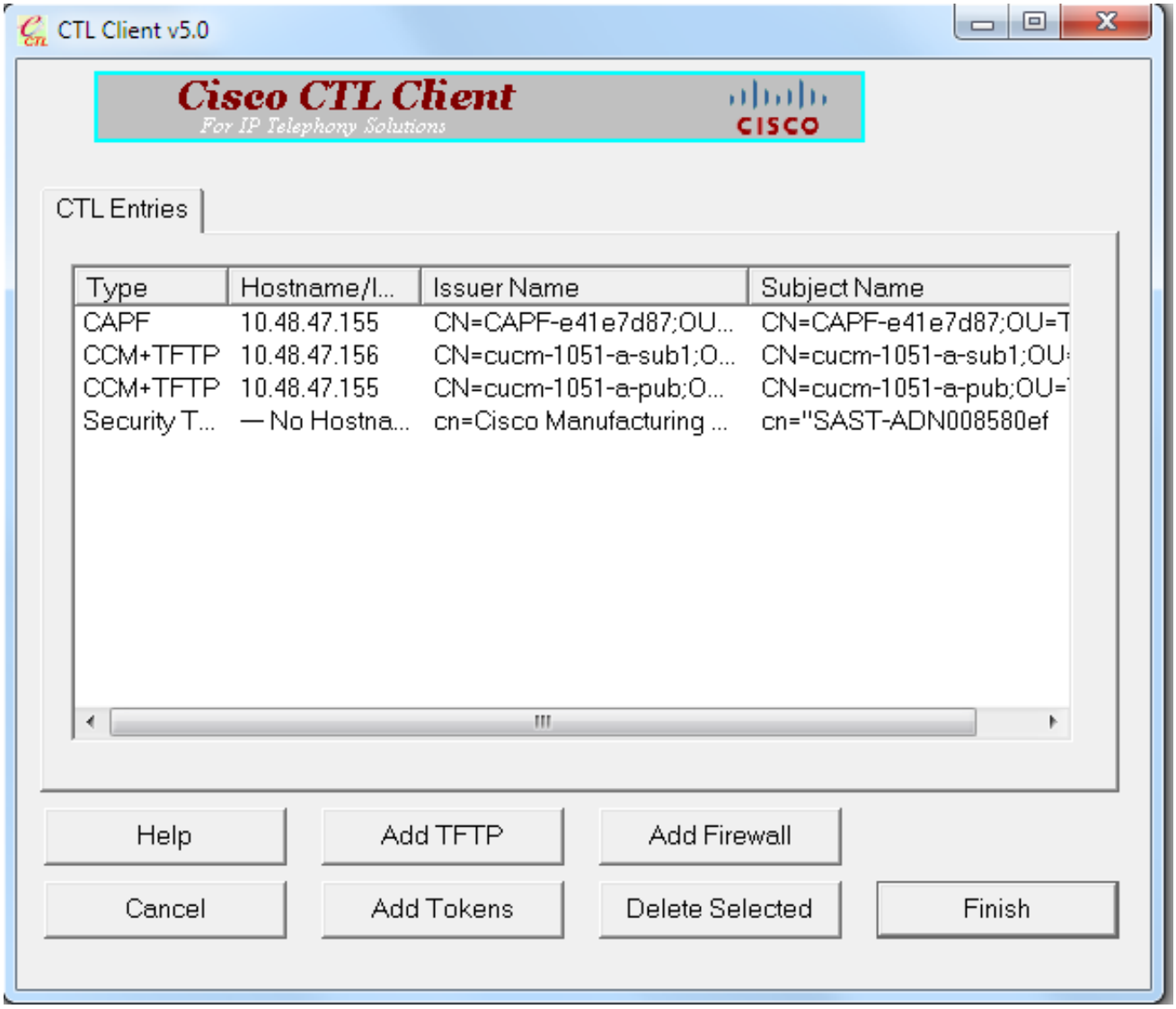
14. Inserte el primer token de seguridad y haga clic en **Aceptar**:



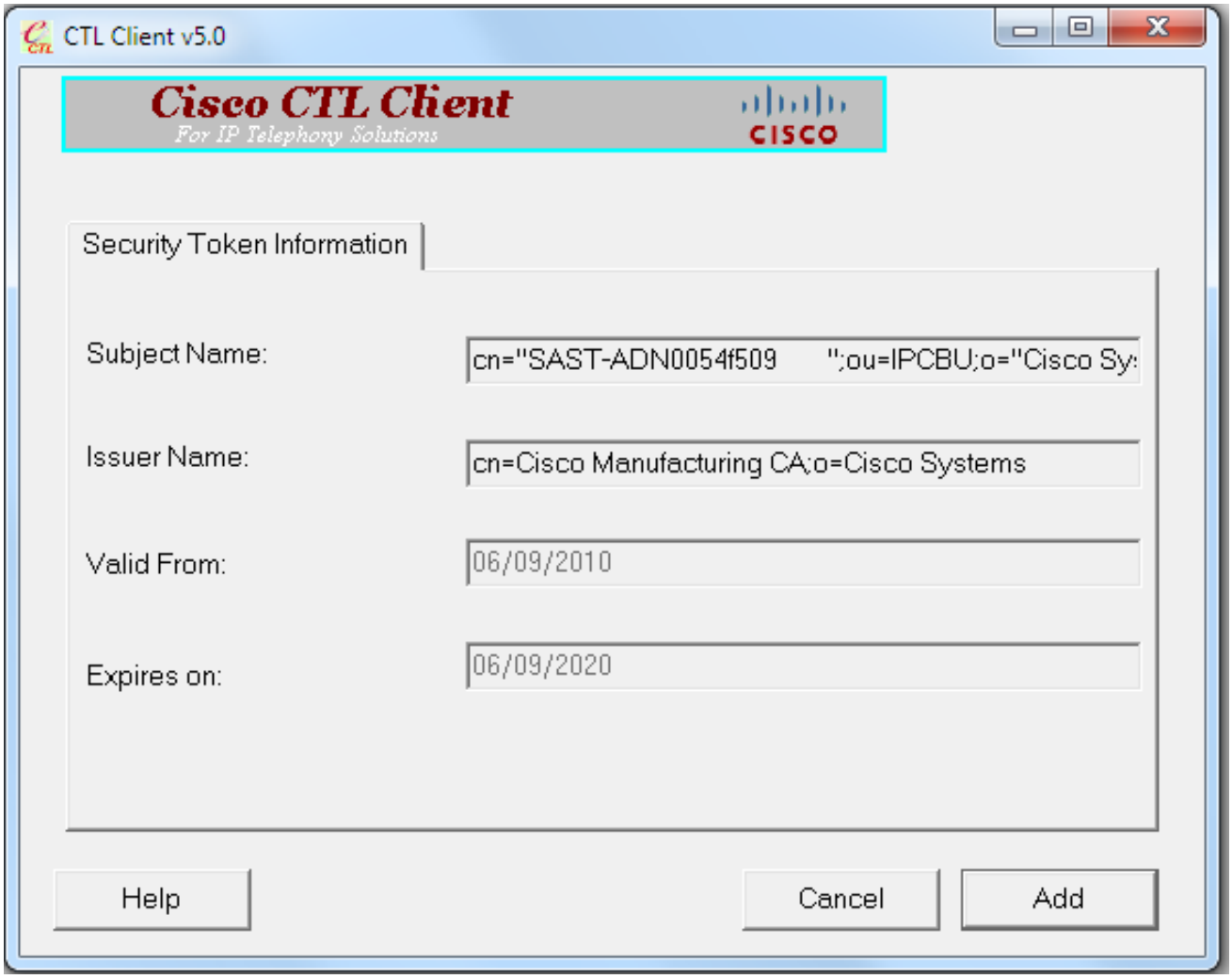
15. Una vez que se muestren los detalles del token de seguridad, haga clic en **Agregar**:



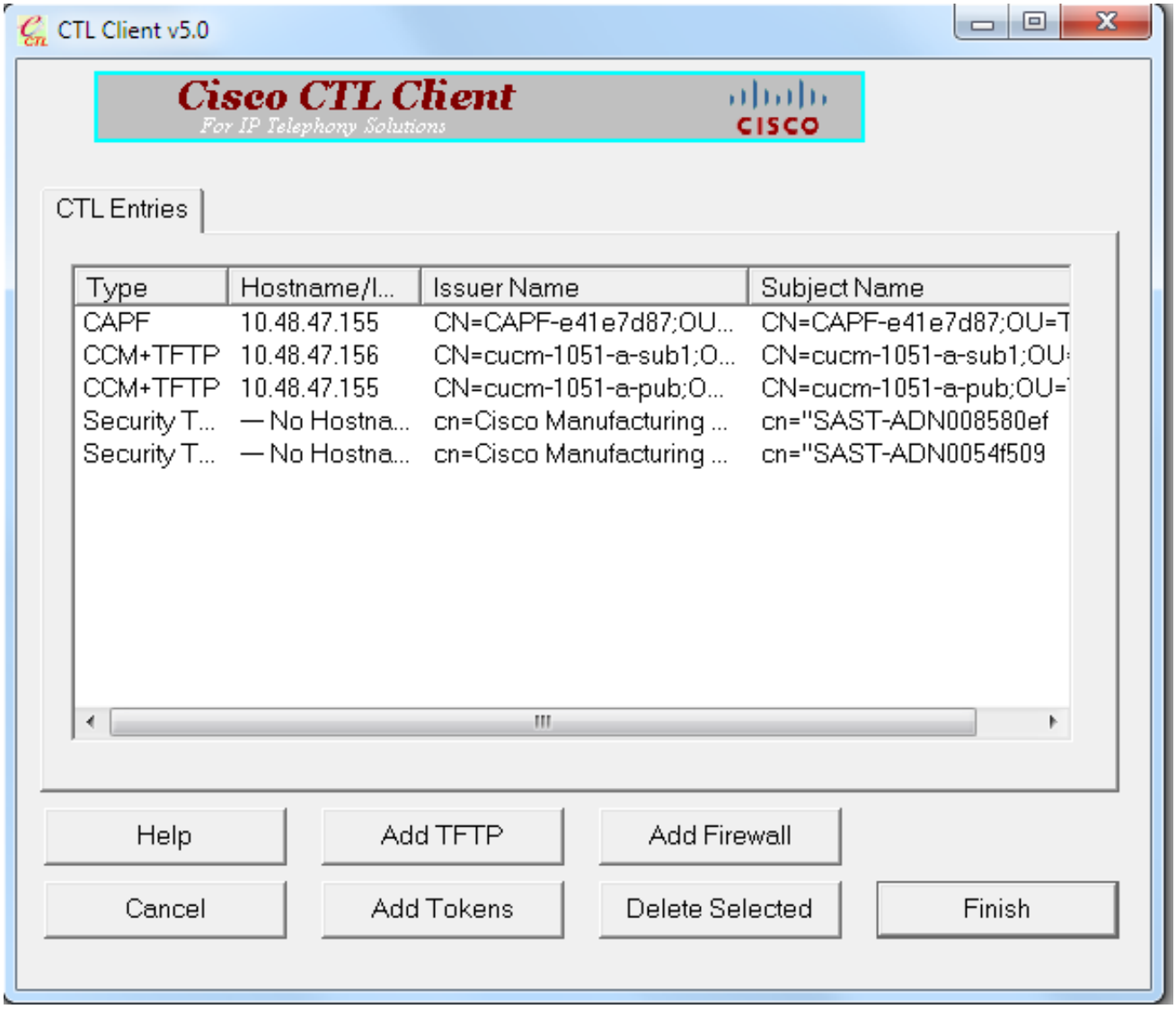
16. Una vez que aparezca el contenido del archivo CTL, haga clic en **Agregar tokens para agregar el segundo eToken USB:**



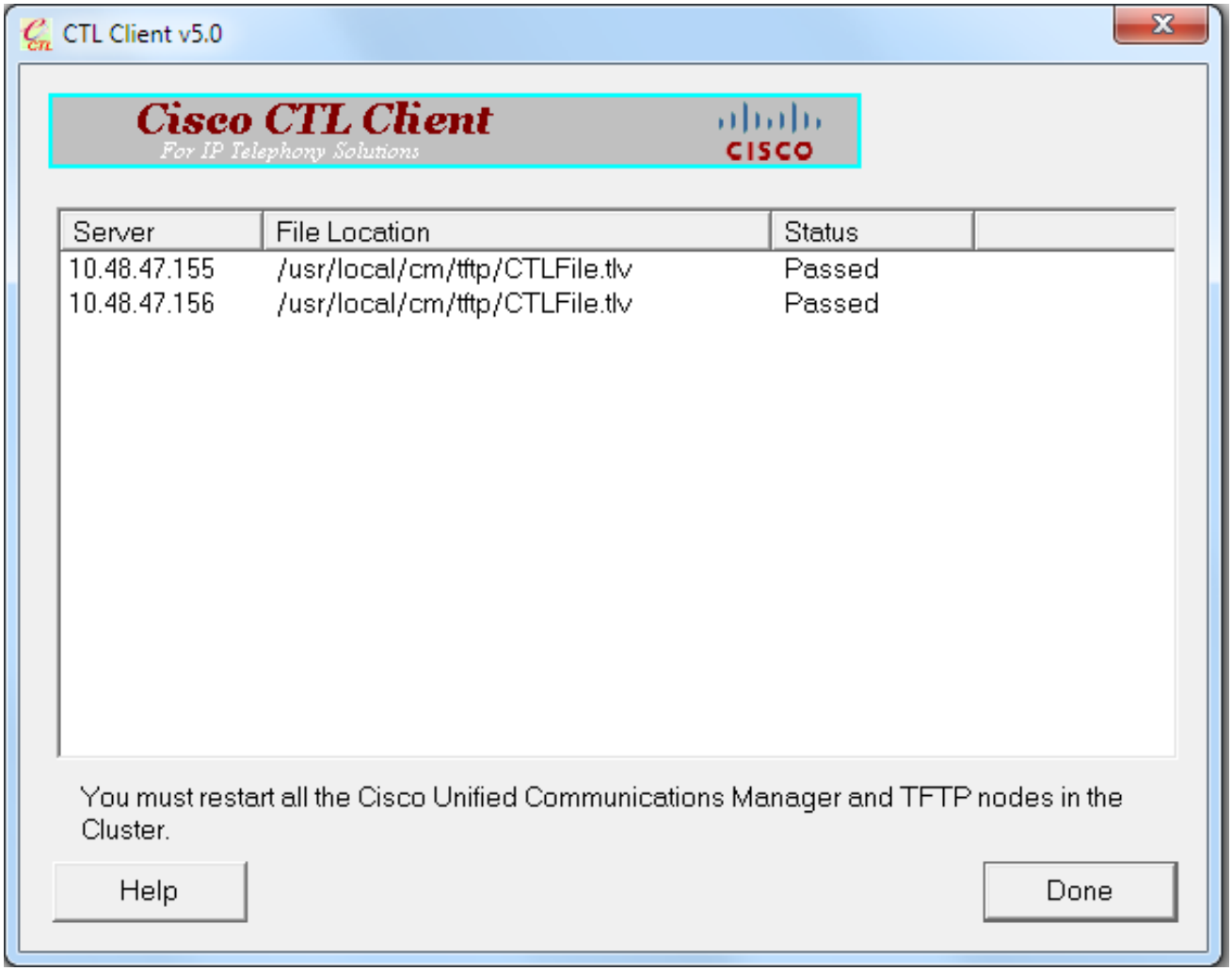
17. Una vez que aparezcan los detalles del token de seguridad, haga clic en **Agregar**:



18. Una vez que aparezca el contenido del archivo CTL, haga clic en Finalizar. Cuando se le solicite una contraseña, ingrese **Cisco123**:



19. Cuando aparezca la lista de servidores CUCM en la que existe el archivo CTL, haga clic en **Listo**:



20. Reinicie los servicios de TFTP y CallManager en todos los nodos del clúster que ejecutan estos servicios.

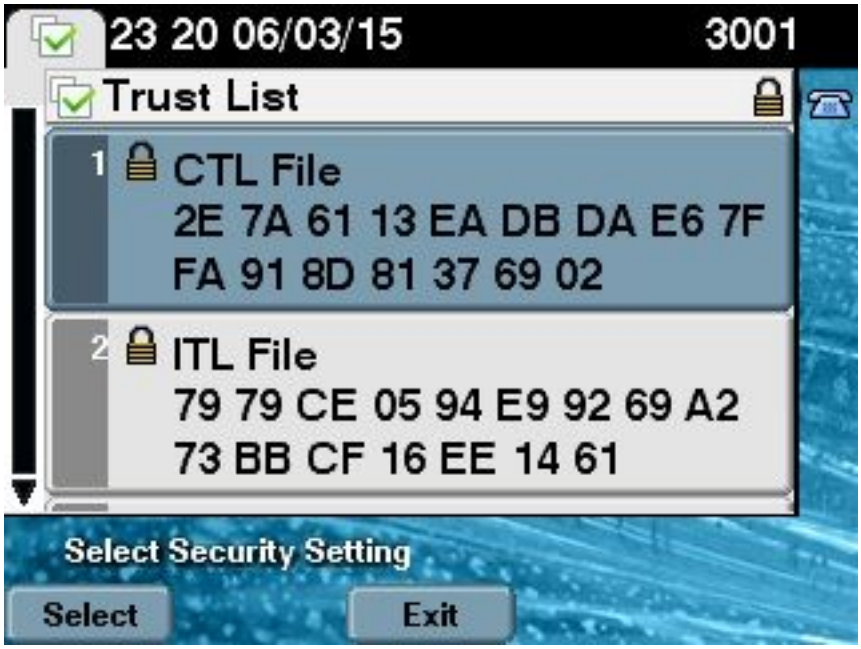
21. Reinicie todos los teléfonos IP para que puedan obtener la nueva versión del archivo CTL del servicio TFTP de CUCM.

22. Para verificar el contenido del archivo CTL, introduzca el comando **show CTL en la CLI.** En el archivo CTL, puede ver los certificados de los dos eTokens USB (uno de ellos se utiliza para firmar el archivo CTL). Éste es un ejemplo de salida:

```
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. En el lado del teléfono IP, puede verificar que, después de que se reiniciaron los teléfonos IP, descargaron la versión actualizada del archivo CTL (las coincidencias de checksum MD5 comparadas con el resultado del CUCM):



Este cambio es posible porque anteriormente exportó y cargó los certificados eToken en el almacén de confianza del certificado CUCM, y los teléfonos IP pueden verificar este certificado desconocido que se utilizó para firmar el archivo CTL en relación con el Servicio de verificación de confianza (TVS) que se ejecuta en la CUCM. Este snippit de registro ilustra la forma en que el teléfono IP se pone en contacto con el TVS de CUCM con una solicitud para verificar el certificado de eToken desconocido, que se carga como **Phone-SAST-trust y es confiable:**

```
//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
```

Regeneración de certificado para la solución CTL sin tokens

En esta sección, se describe cómo regenerar un certificado de seguridad de clúster de CUCM cuando se utiliza la solución de CTL sin tokens.

En el proceso de mantenimiento de CUCM, algunas veces el certificado CallManager del nodo de editor de CUCM cambia. Entre las situaciones en las que esto puede suceder, se incluye el cambio de nombre de host, el cambio de dominio o simplemente una regeneración de certificado (debido a la proximidad de la fecha de vencimiento del certificado).

Una vez que se actualiza el archivo CTL, se firma con un certificado diferente al de los que existen en el archivo CTL que se instala en los teléfonos IP. Por lo general, este nuevo archivo CTL no se acepta; sin embargo, después de que el teléfono IP encuentra el certificado desconocido que se utiliza para firmar el archivo CTL, se pone en contacto con el servicio de TVS en CUCM.

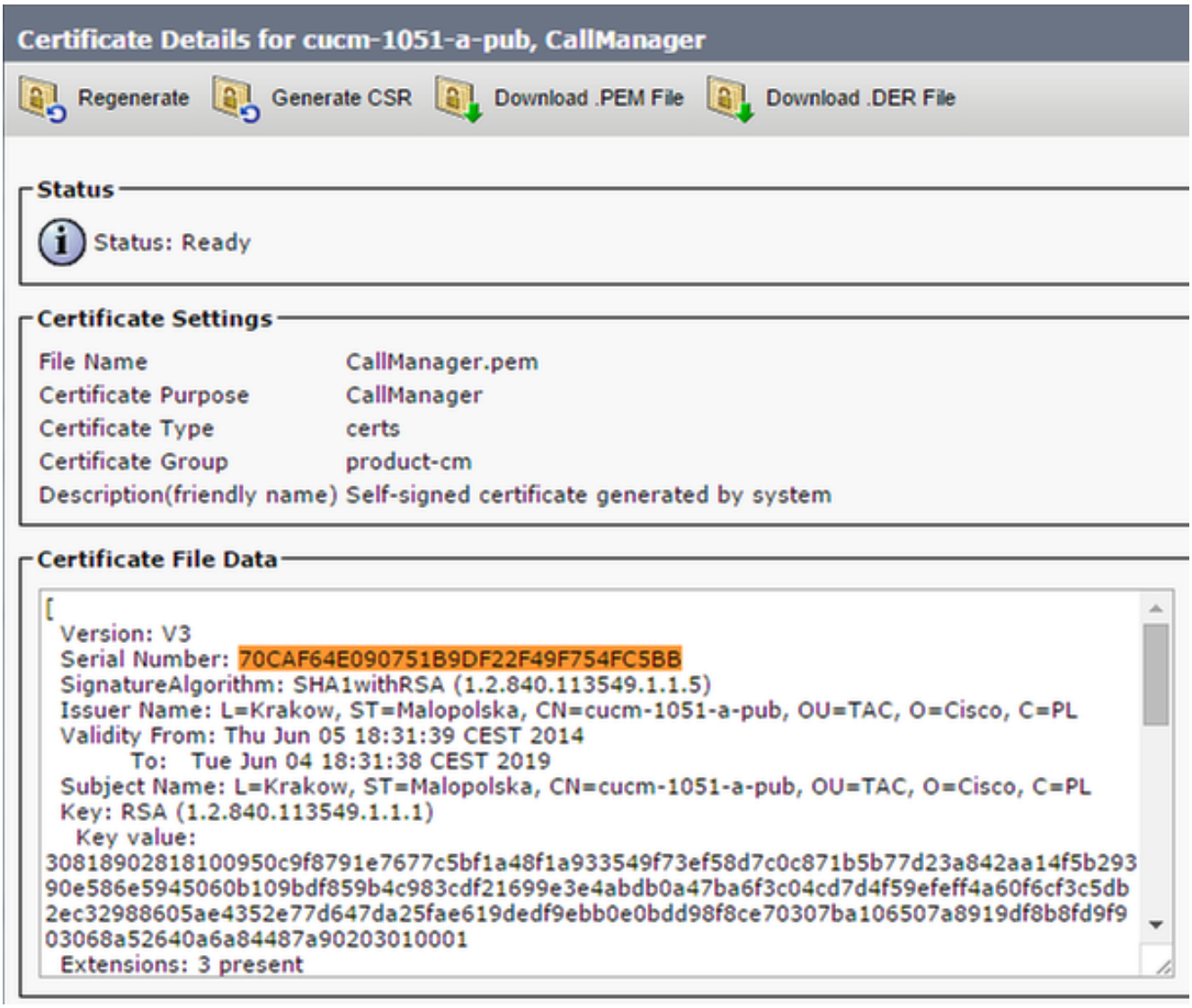
Nota: La lista de servidores de TVS se encuentra en el archivo de configuración del teléfono IP y se asigna en los servidores de CUCM desde **Grupo de dispositivos > Grupo de CallManager** del teléfono IP.

Una vez que la verificación con el servidor TVS se realizó con éxito, el teléfono IP actualiza su archivo CTL con la nueva versión. Estos eventos se producen en una situación como la siguiente:

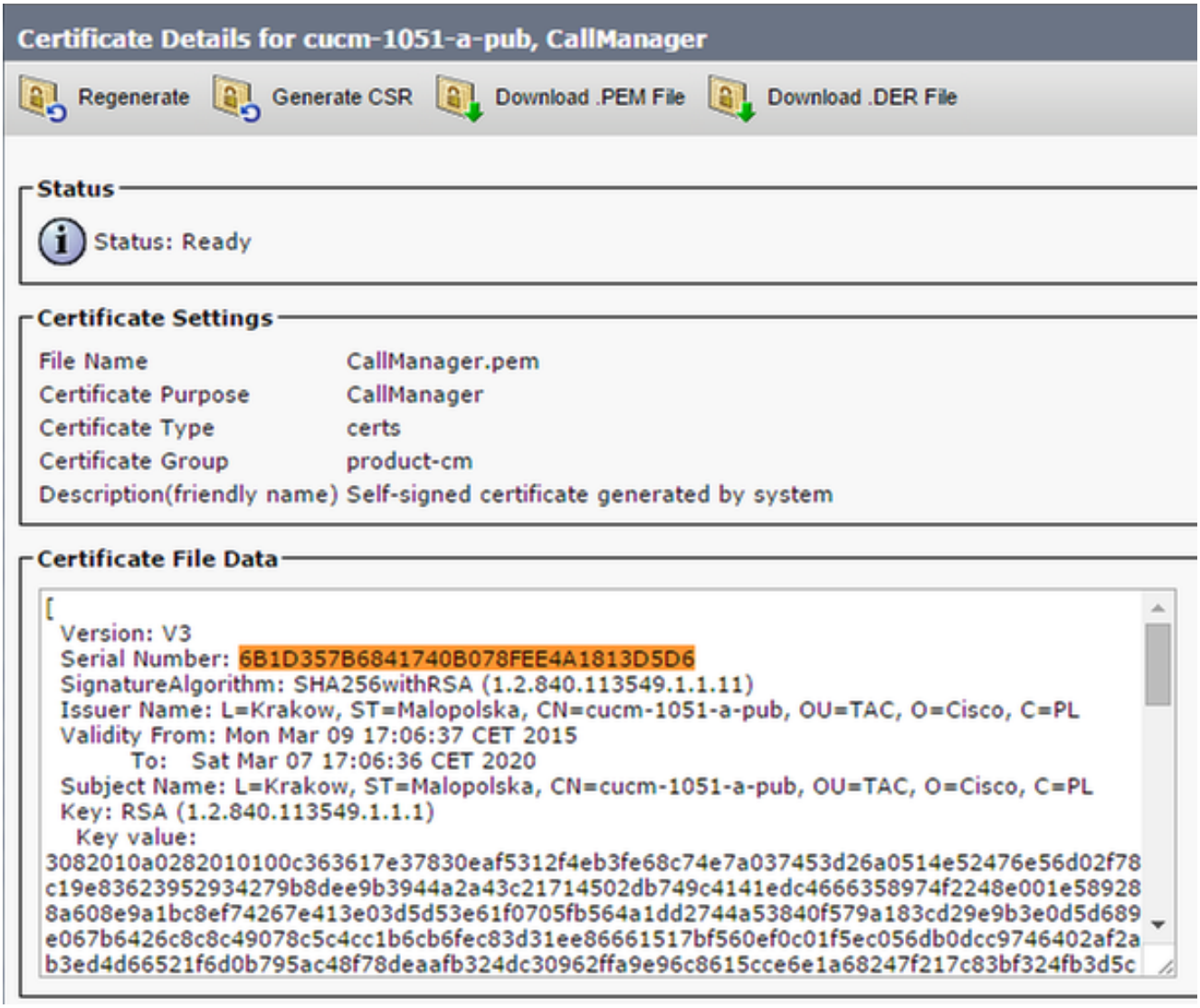
1. El archivo CTL existe en el CUCM y en el teléfono IP. El certificado de CCM+TFT (servidor) para el nodo de editor de CUCM se utiliza para firmar el archivo CTL:

```
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. Se regenera el archivo CallManager.pem (certificado CCM+TFTP) y puede ver que el número de serie del certificado cambia:



3. El comando **utils ctl update CTLFile se introduce en la CLI para actualizar el archivo 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. El servicio de TVS actualiza su caché de certificados con los nuevos detalles del archivo 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. Cuando vea el contenido del archivo CTL, podrá ver que el archivo está firmado con el nuevo certificado del servidor CallManager para el nodo de editor:

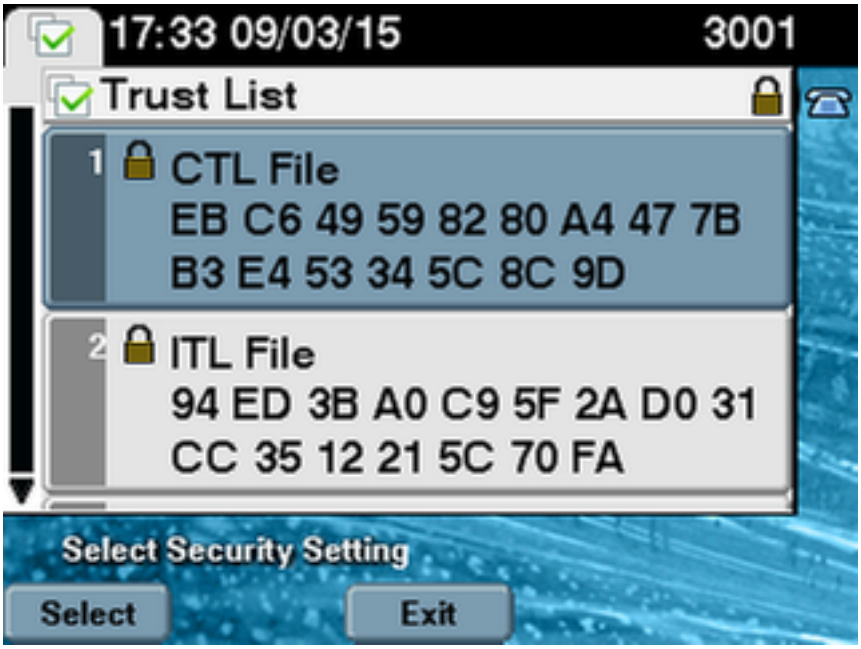
```
admin:show ctl
The checksum value of the CTL file:
ebc649598280a4477bb3e453345c8c9d(MD5)
ef5c006b6182cad66197fac6e6530f15d009319d(SHA1)

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. De la página Unified Serviceability, el TFTP y los servicios de Cisco CallManager se reinician en todos los nodos del clúster que ejecutan estos servicios.

7. Los teléfonos IP se reinician y se comunican con el servidor TVS para verificar el certificado desconocido que ahora se usa para firmar la nueva versión del archivo 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. Por último, en los teléfonos IP, puede verificar que el archivo CTL esté actualizado con la nueva versión y que la checksum MD5 del nuevo archivo CTL coincida con la de la CUCM:


17:33 09/03/15
3001
Trust List
1
CTL File
EB C6 49 59 82 80 A4 47 7B
B3 E4 53 34 5C 8C 9D
2
ITL File
94 ED 3B A0 C9 5F 2A D0 31
CC 35 12 21 5C 70 FA
Select Security Setting
Select
Exit