Modo combinado de CUCM com CTL sem token

Contents

[Introduction](#)
[Prerequisites](#)
[Requirements](#)
[Componentes Utilizados](#)
[Informações de Apoio](#)
[Do modo não seguro para o modo misto (CTL sem token)](#)
[De eTokens de hardware para uma solução sem token](#)
[Da solução sem token para eTokens de hardware](#)
[Geração nova de certificado para solução CTL sem token](#)

Introduction

Este documento descreve a diferença entre a segurança do Cisco Unified Communications Manager (CUCM) com e sem o uso de eTokens USB de hardware. Este documento também descreve os cenários de implementação básica que envolvem a lista de certificados confiáveis sem token (CTL) e o processo usado para garantir que o sistema funcione corretamente após as alterações.

Prerequisites

Requirements

A Cisco recomenda que você tenha experiência com o CUCM versão 10.0 (1) ou posterior. Além disso, verifique se:

- O servidor de licenças para CUCM versão 11.5.1SU3 e posterior deve ser Cisco Prime License Manager (PLM) 11.5.1 SU2 ou posterior. Isso se deve ao fato de que o CUCM versão 11.5.1 SU3 requer a licença de criptografia para ativar o modo misto, e o PLM não é compatível com a licença de criptografia até a SU2 11.5.1. Para obter mais informações, consulte as Notas de versão do Cisco Prime License Manager, versão 11.5(1) SU2.

- Você tem acesso administrativo à interface de linha de comando (CLI) do nó do editor CUCM.

- Você tem acesso a eTokens USB de hardware e o plug-in do cliente CTL está instalado no PC para cenários que exigem a migração de volta ao uso de eTokens de hardware. Para maior clareza, esse requisito é apenas para um cenário em que os eTokens USB são necessários. As chances são muito pequenas de que os eTokens USB sejam necessários para a maioria das pessoas.

- Há conectividade total entre todos os nós de CUCM no cluster. Isso é muito importante porque o arquivo CTL é copiado para todos os nós no cluster através do SSH File Transfer

Protocol (SFTP).

- A replicação do banco de dados (BD) no cluster funciona corretamente e os servidores replicam os dados em tempo real.

- Os dispositivos na implantação dão suporte à segurança por padrão (TVS). Você pode usar a *Unified CM Phone Feature List (Lista de recursos de telefone CM unificado) na página da Web de Cisco Unified Reporting (https://<CUCM IP or FQDN>/cucreports/) para determinar os dispositivos que oferecem suporte à segurança por padrão.*

  Note: O Cisco Jabber e muitos telefones IP Cisco TelePresence ou Cisco 7940/7960 Series não oferecem suporte atualmente à segurança por padrão. Se você implantar CTL sem token com dispositivos que não respaldam a segurança por padrão, qualquer atualização do sistema que altere o certificado CallManager no editor impedirá a funcionalidade normal desses dispositivos, até que a CTL seja excluída manualmente. Os dispositivos compatíveis com a segurança por padrão, como os telefones 7945 e 7965 ou mais recentes, podem instalar arquivos CTL quando o certificado CallManager no editor é atualizado, pois eles conseguem usar o Trust Verification Service (TVS).

Componentes Utilizados

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

- CUCM versão 10.5.1.10000-7 (cluster de dois nós)

- Telefones IP Cisco 7975 Series registrados via Skinny Client Control Protocol (SCCP) com Firmware versão SCCP 75.9-3-1SR4-1S

- Dois tokens de segurança da Cisco usados para definir o cluster como modo misto com o uso do software cliente CTL

Informações de Apoio

CTL sem token é um novo recurso nas versões do CUCM 10.0(1) e posterior que possibilita a criptografia de sinalização de chamadas e de mídia para telefones IP, sem a necessidade de usar eTokens USB de hardware e o plug-in do cliente CTL, que foi o requisito nas versões anteriores do CUCM.

Quando o cluster é colocado no modo misto com o uso do comando CLI, o arquivo CTL é assinado com o certificado CCM+TFTP (Servidor) do nó do editor e não há certificados eToken presentes no arquivo CTL.

Note: Quando você gera novamente o certificado CallManager (CCM+TFTP) no editor, ele altera o signatário do arquivo. Os telefones e dispositivos não compatíveis com segurança por padrão não aceitarão o novo arquivo CTL, a menos que os arquivos CTL sejam excluídos manualmente de cada dispositivo. Consulte a última exigência listada na seção Requisitos deste documento para obter mais informações.

Do modo não seguro para o modo misto (CTL sem token)

Esta seção descreve o processo usado para mover a segurança do cluster de CUCM para o modo misto via CLI.

Antes desse cenário, o CUCM estava no modo não seguro, o que significa que não havia um arquivo CTL presente em nenhum dos nós e que os telefones IP registrados tinham apenas um arquivo de lista de confiança de identidade (ITL) instalado, como mostrado nestas 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:
```

> **Note**: Se for encontrado um arquivo CTL no servidor enquanto o cluster não estiver no modo misto, isso significa que o cluster estava no modo misto e, em seguida, foi transferido de volta para o modo não misto, e o arquivo CTL não foi excluído do cluster.
>
> O arquivo de comando delete activelog cm/tftpdata/CTLFile.tlv excluio arquivo CTL dos nós no cluster CUCM; no entanto, o comando precisa ser inserido em cada nó. Para ser claro, só use esse comando se os servidores tiverem um arquivo CTL e o cluster não estiver no modo misto.
>
> Uma maneira fácil de confirmar se um cluster está no modo misto é usar o comando **run sql select paramname,paramvalue from processconfig onde paramname='ClusterSecurityMode'**. Se o valor de param é 0, o cluster não está no modo misto.

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

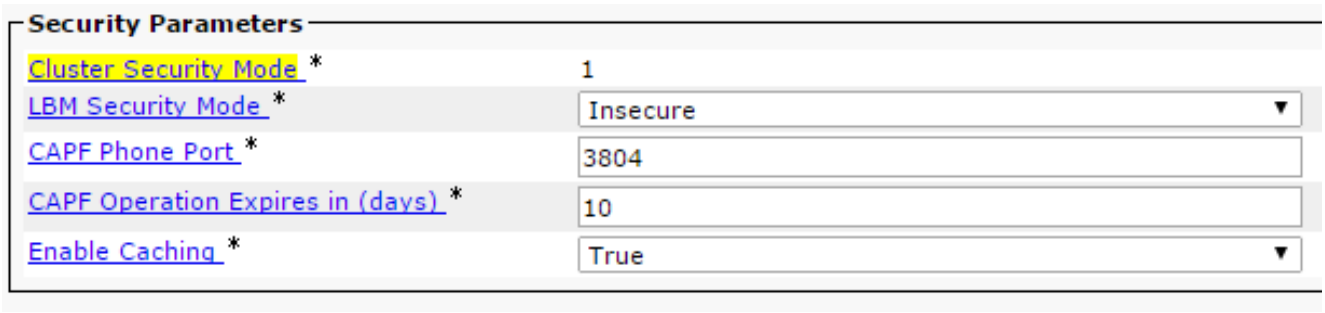
Para transferir a segurança de cluster do CUCM para o modo misto com o uso do novo recurso de CTL sem token, siga estas etapas:

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

2. Insira o comando **utils ctl set-cluster mixed-mode 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 até **CUCM Admin Page > System > Enterprise Parameters** (Página de administrador CUCM > Sistema > Parâmetros corporativos) e verifique se o cluster foi definido como o modo Mixed (Misto) (um valor **1 indica o modo misto):**



4. Reinicie os serviços TFTP e Cisco CallManager em todos os nós no cluster que os executam.

5. Reinicie todos os telefones IP para que eles possam obter o arquivo CTL do serviço TFTP CUCM.

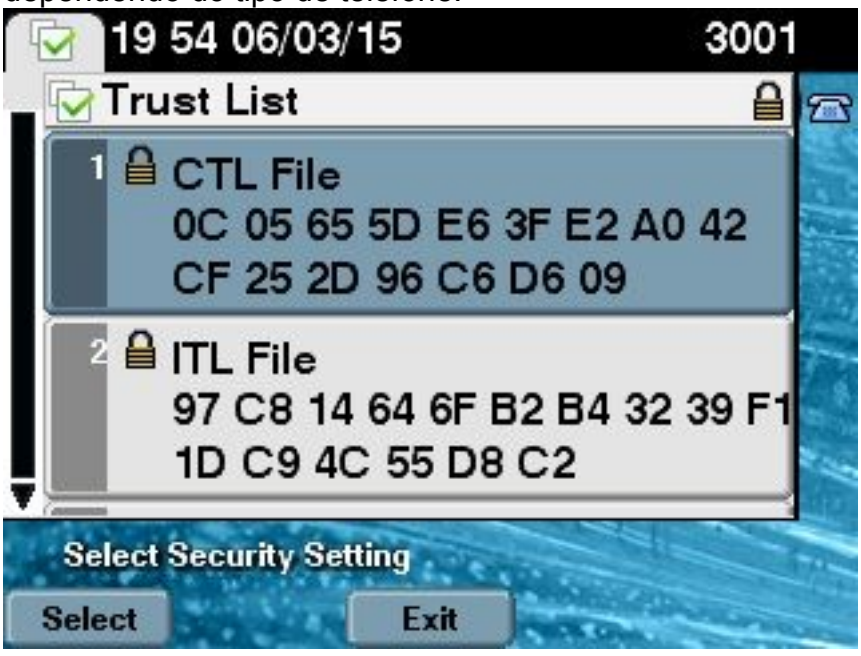
6. Para verificar o conteúdo do arquivo CTL, insira o comando **show ctl na CLI.** No arquivo CTL, você pode ver que o certificado CCM+TFTP (servidor) para o nó do editor CUCM é usado para assinar o arquivo CTL (esse arquivo é o mesmo em todos os servidores no cluster). Veja um exemplo de saída:

```
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 se depois que o serviço é reiniciado, ele baixa o arquivo CTL, presente no servidor TFTP (as correspondências da soma de verificação MD5 são comparadas à saída do CUCM):

   Observação: ao verificar a soma de verificação no telefone, você verá **MD5** ou **SHA1**, dependendo do tipo de telefone.



De eTokens de hardware para uma solução sem token

Esta seção descreve como migrar a segurança de cluster CUCM dos eTokens de hardware para o uso da nova solução sem token.

Em algumas situações, o modo misto já está configurado no CUCM com o uso do cliente de CTL, e os telefones IP usam arquivos CTL com certificados de eTokens de USB de hardware. Com esse cenário, o arquivo CTL é assinado por um certificado de um dos USB de eTokens e é instalado nos telefones IP. Aqui há 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.
```

Conclua essas etapas para transferir a segurança de cluster do CUCM e usar os CTLs sem token:

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

2. Insira o comando da CLI **utils ctl update CTLFile:**

```
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 os serviços TFTP e CallManager em todos os nós no cluster que os executam.

4. Reinicie todos os telefones IP para que eles possam obter o arquivo CTL do serviço TFTP CUCM.

5. Para verificar o conteúdo do arquivo CTL, insira o comando **show ctl** na CLI. No arquivo CTL, você pode ver que o certificado CCM+TFTP (servidor) do nó do editor CUCM é usado para assinar o arquivo CTL em vez do certificado de eTokens de USB de hardware. Uma diferença mais importante nesse caso é que os certificados de todos os eTokens de USB de hardware são removidos do arquivo CTL. Veja 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.
```

Note: Na saída acima, se o certificado CCM+TFTP (servidor) do CUCM Publisher não for signatário, volte para o modo de segurança de cluster baseado em etoken de hardware e repita as alterações para a solução sem token.

6. No lado do telefone IP, você pode verificar se depois de baixar os telefones IP, ele baixa a versão atualizada do arquivo CTL (as correspondências da soma de verificação MD5 são comparadas à saída do CUCM):

Da solução sem token para eTokens de hardware

Esta seção descreve como migrar a segurança de cluster CUCM para longe da nova solução sem token e voltar ao uso de eTokens de hardware.

Quando a segurança de cluster de CUCM é colocada no modo misto com o uso do comandos CLI, e o arquivo CTL é assinado com o certificado CCM+TFTP (Servidor) para o nó do editor, não há certificados de eToken de USB de hardware presentes no arquivo CTL. Por esse motivo, quando você executa o cliente CTL para atualizar o arquivo CTL (voltar ao uso de hardware eTokens), essa mensagem de erro é exibida:

```
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.
```

Isso é especialmente importante em cenários que incluem um downgrade (quando a versão é revertida) do sistema para uma versão anterior a 10.x que não inclui os comandos **utils clt**. O arquivo CTL anterior é migrado (sem alterações no conteúdo) no processo de uma atualização ou um upgrade de Linux para Linux (L2) e não contém os certificados de eToken, como mencionado anteriormente. Veja 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:
```

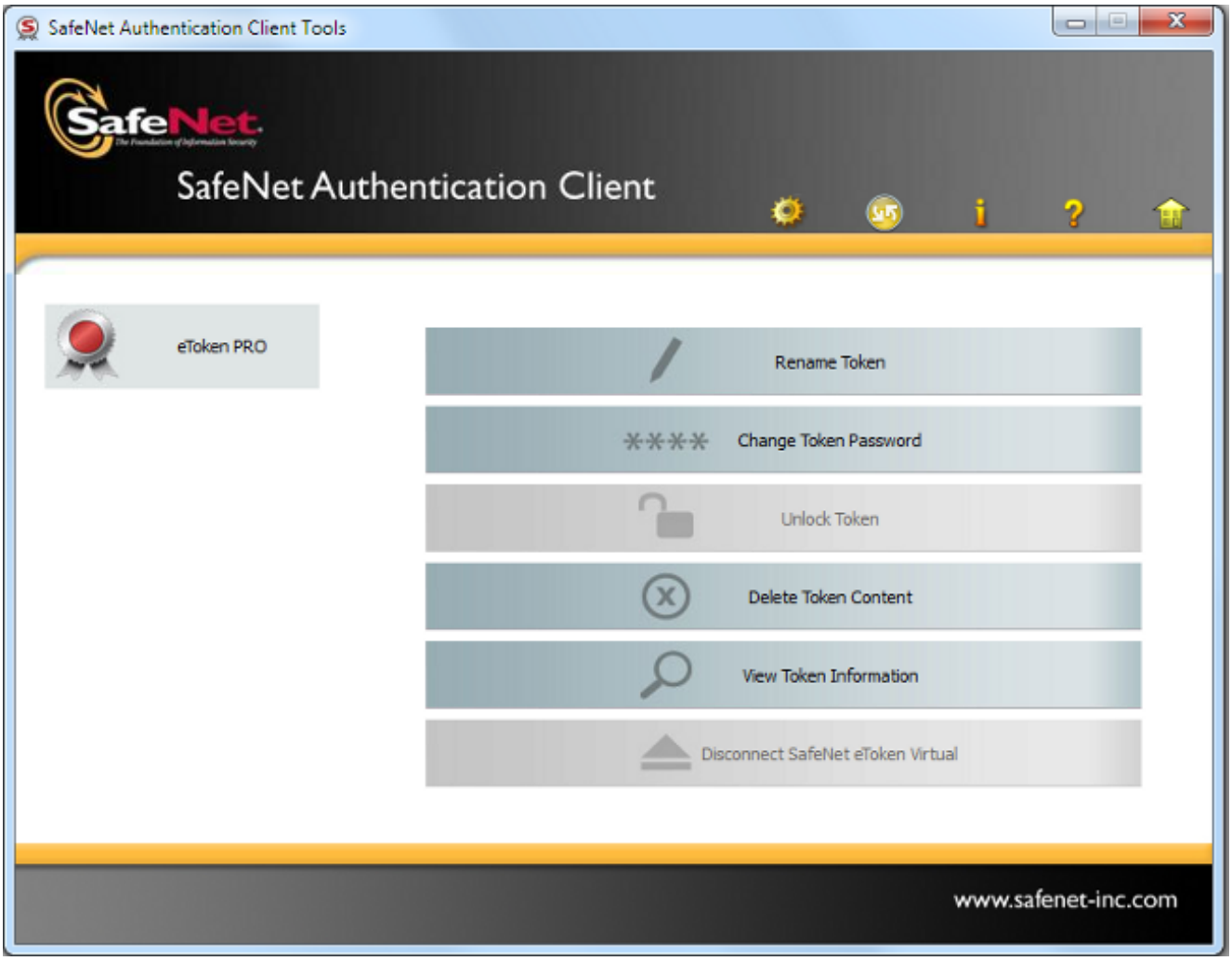
Para esse cenário, siga estas etapas para atualizar com segurança os arquivos CTL sem a necessidade de usar o procedimento para perda de eTokens, que acaba na exclusão manual do arquivo CTL de todos os telefones IP:

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

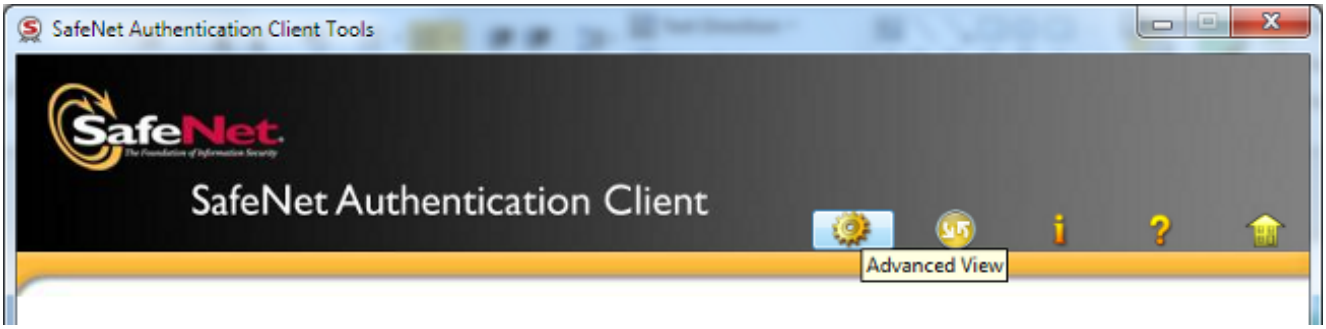
2. Insira o comando **file delete tftp CTLFile.tlv command na CLI do nó do editor para excluir o arquivo CTL:**

```
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 SafeNet na máquina com Microsoft Windows que tem o cliente CTL instalado (ele é instalado automaticamente com o cliente CTL):**

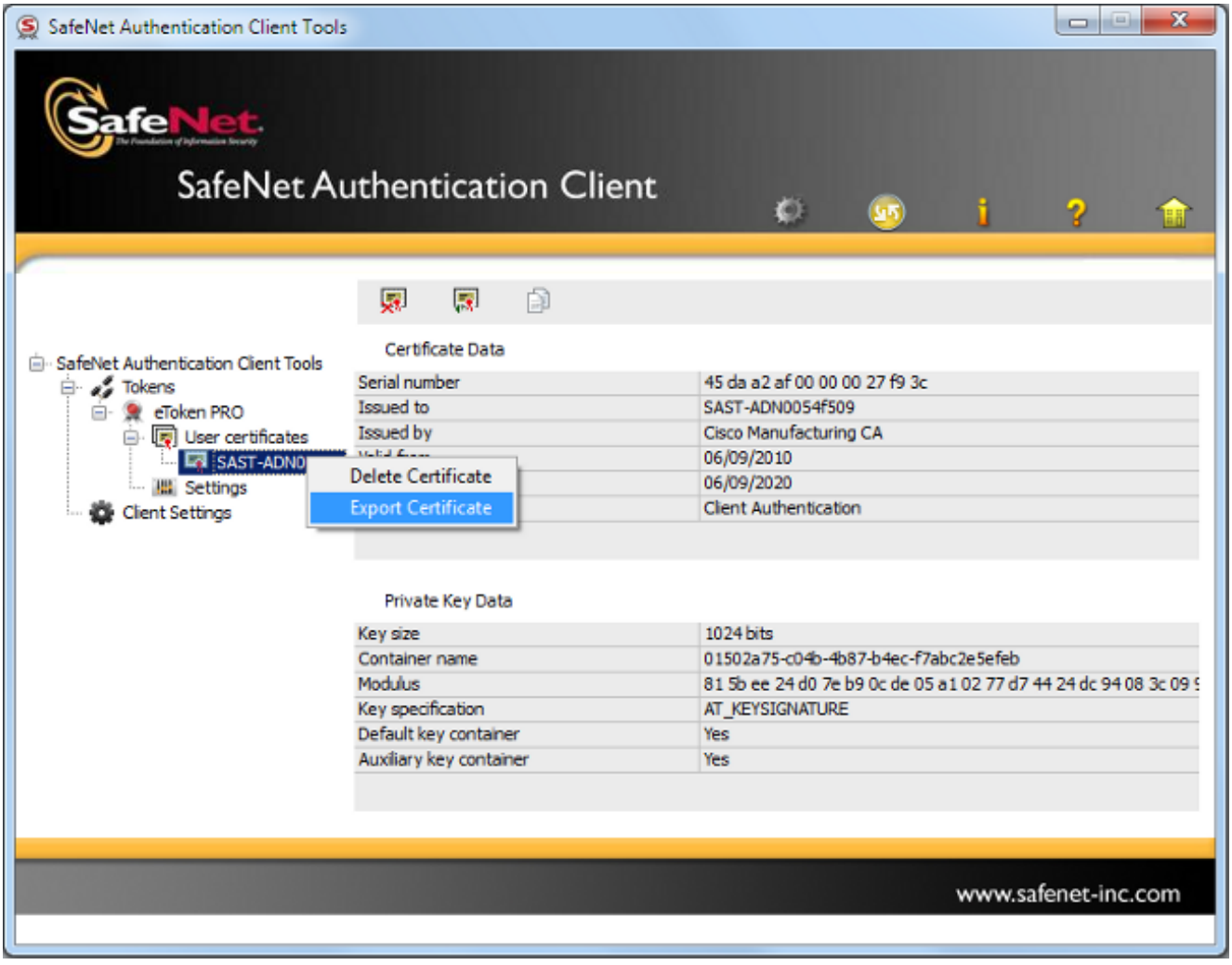


4. No cliente de autenticação SafeNet, navegue até a *Advanced View*(Exibição avançada):

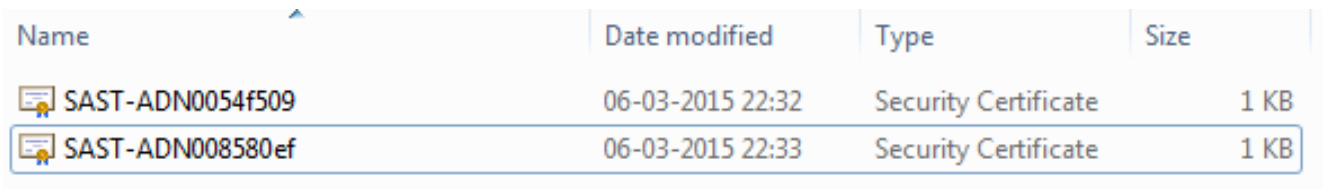


5. Insira o eToken USB de hardware.

6. Selecione o certificado na pasta *User certificates (Certificados de usuário) e exporte-o para a pasta no PC.* Quando solicitada uma senha, use a senha padrão **Cisco123**:

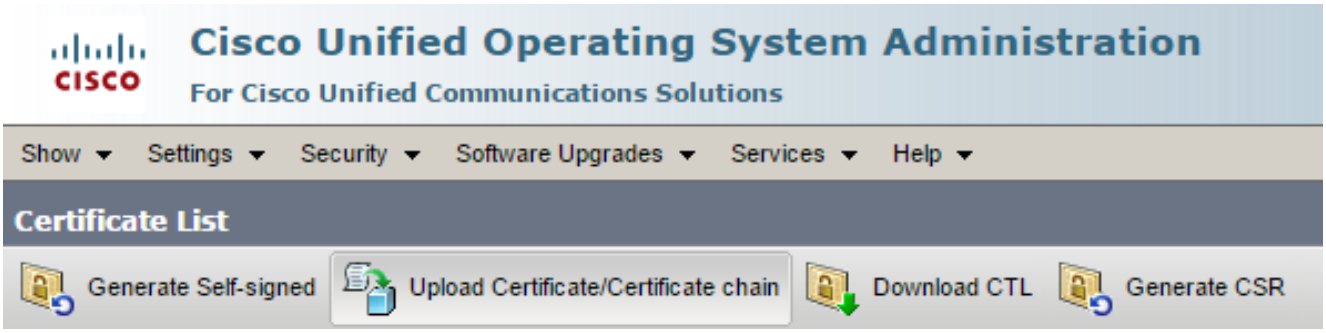


7. Repita essas etapas para o segundo eToken USB de hardware para que os dois certificados sejam exportados para o PC:

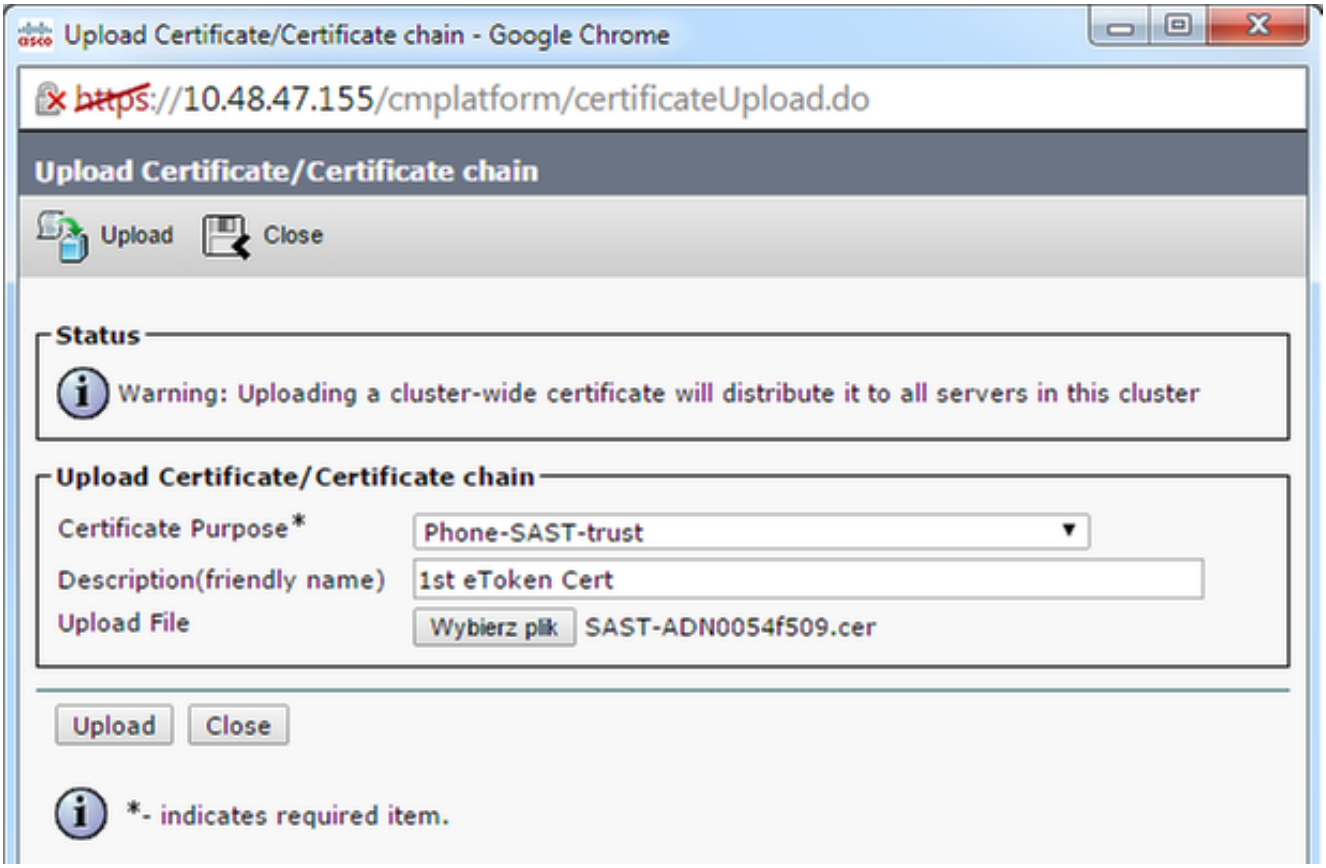


8. Faça login na Administração do sistema operacional Cisco Unified Administration e navegue

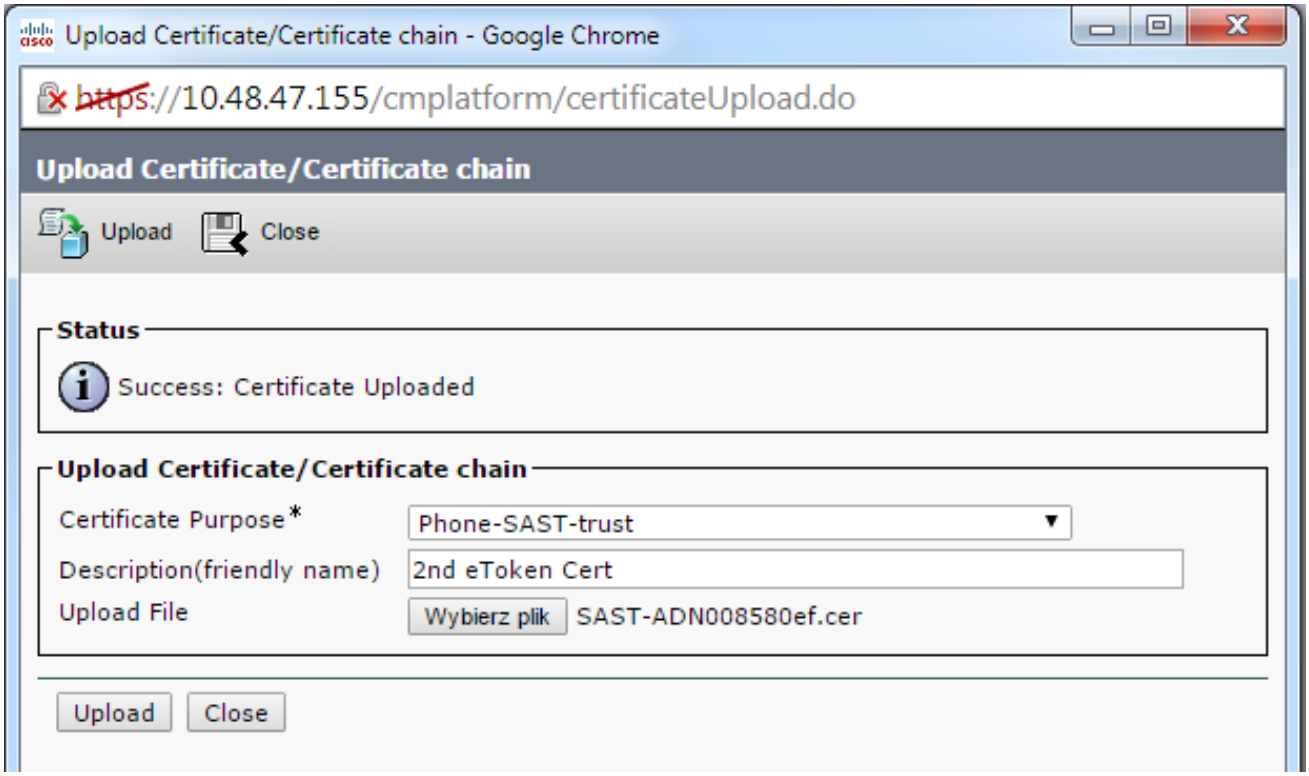
até **Security > Certificate Management > Upload Certificate** (Segurança > Gerenciamento de certificado > Carregar certificado):



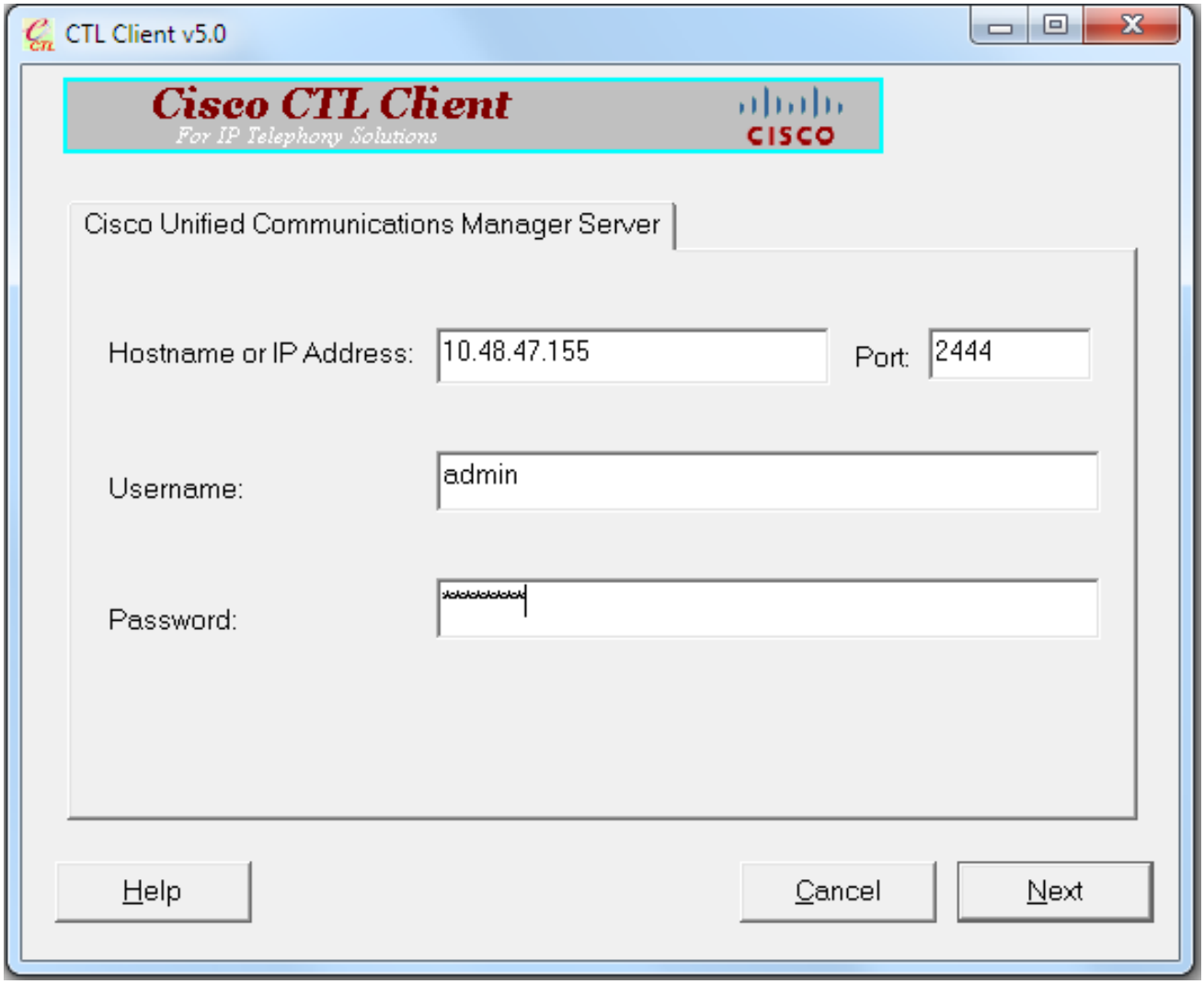
9. A página Upload Certificate (Carregar certificado) será exibida. Escolha **Phone-SAST-trust no menu suspenso de finalidade do certificado e selecione o certificado que você exportou do primeiro eToken:**



10. Conclua as etapas anteriores para carregar o certificado que você exportou do segundo eToken:



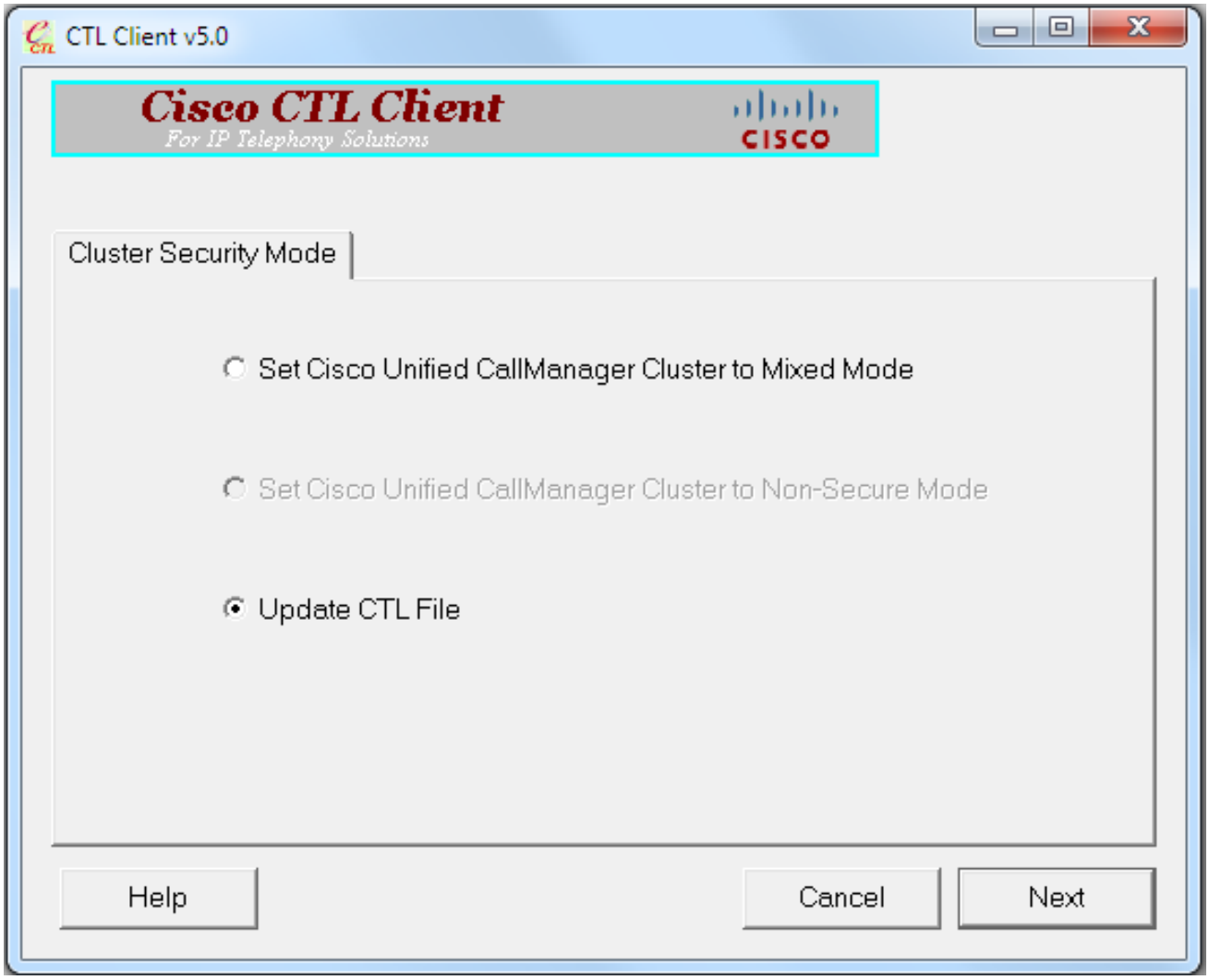
11. Execute o cliente CTL, forneça o endereço IP/nome de host do nó de editor CUCM e insira as credenciais de administrador CCM:



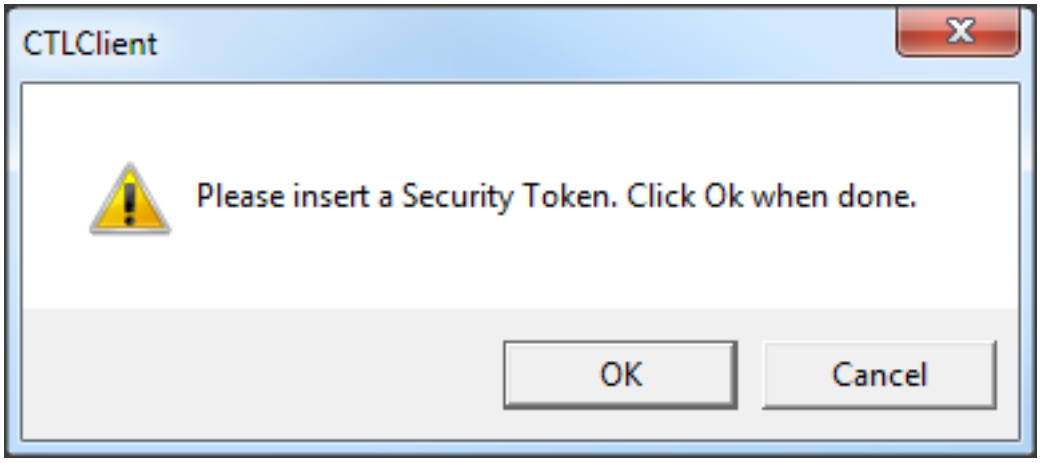
12. Como o cluster já está no modo misto, mas não existe um arquivo CTL no nó do editor, essa mensagem de aviso é (clique em **OK para ignorá- la):**

```
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.
```

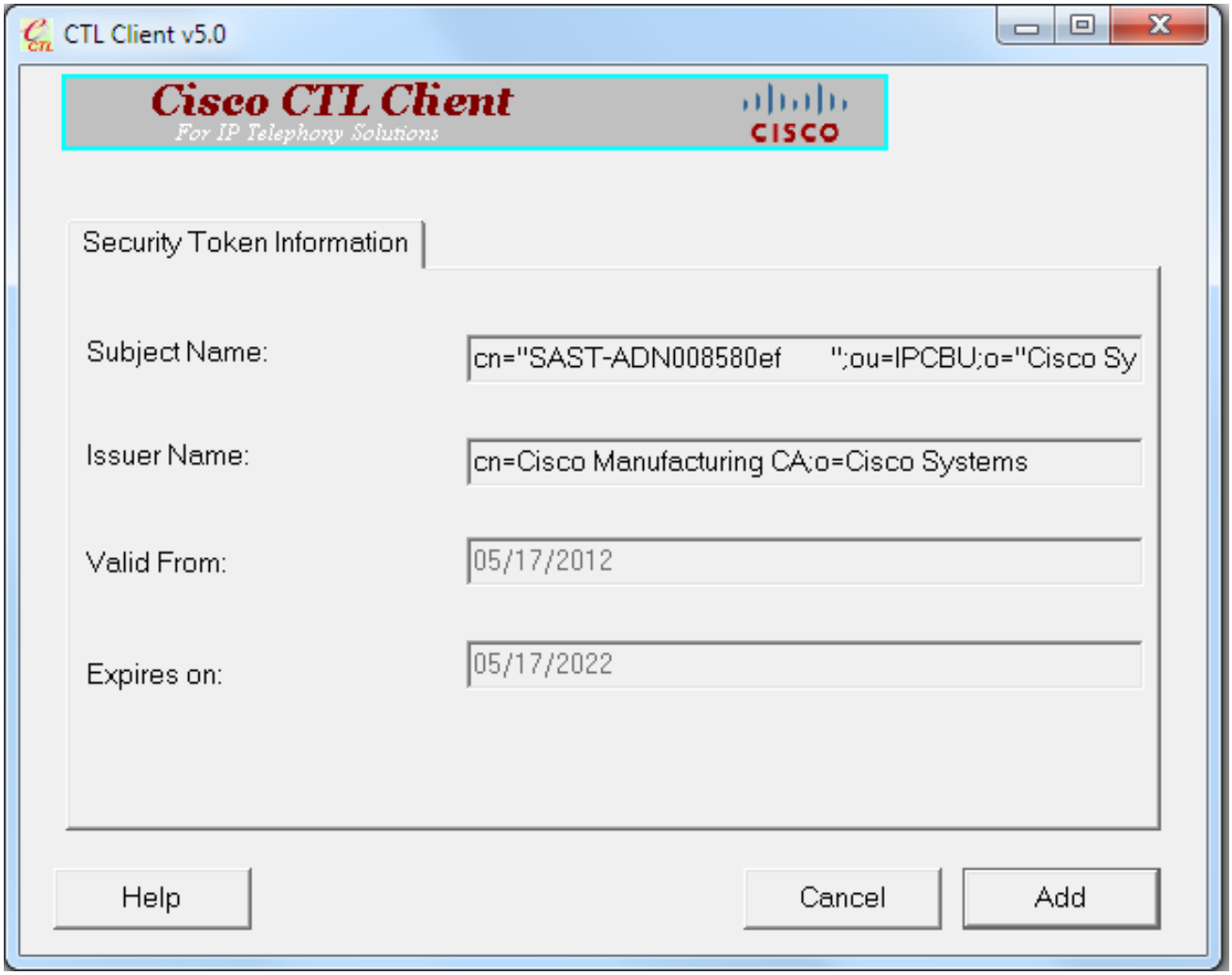
13. No cliente CTL, clique no botão de rádio **Update CTL File (Atualizar o arquivo CTL) e, em seguida, clique em Next** (Avançar):



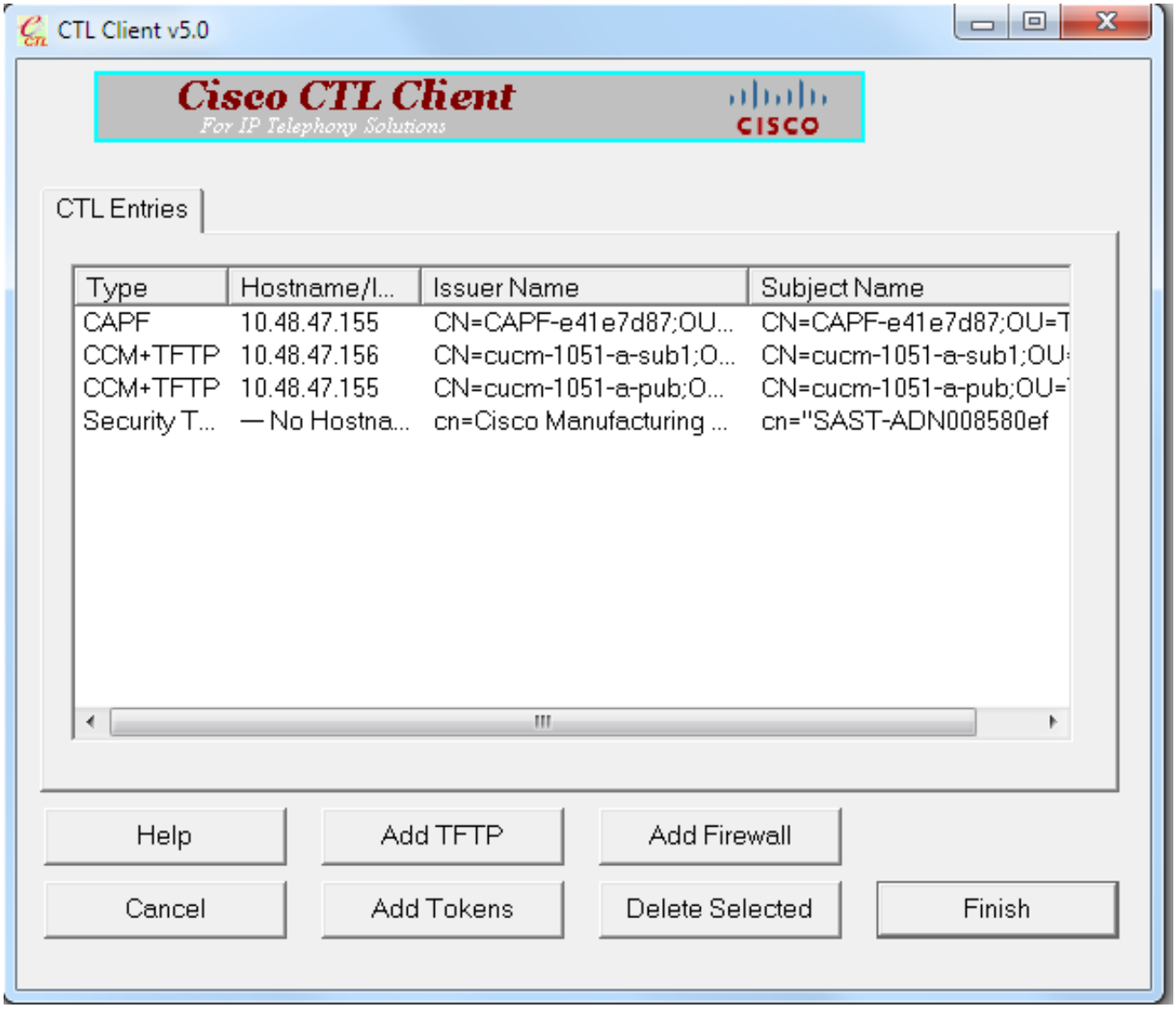
14. Insira o primeiro token de segurança e clique em **OK**:



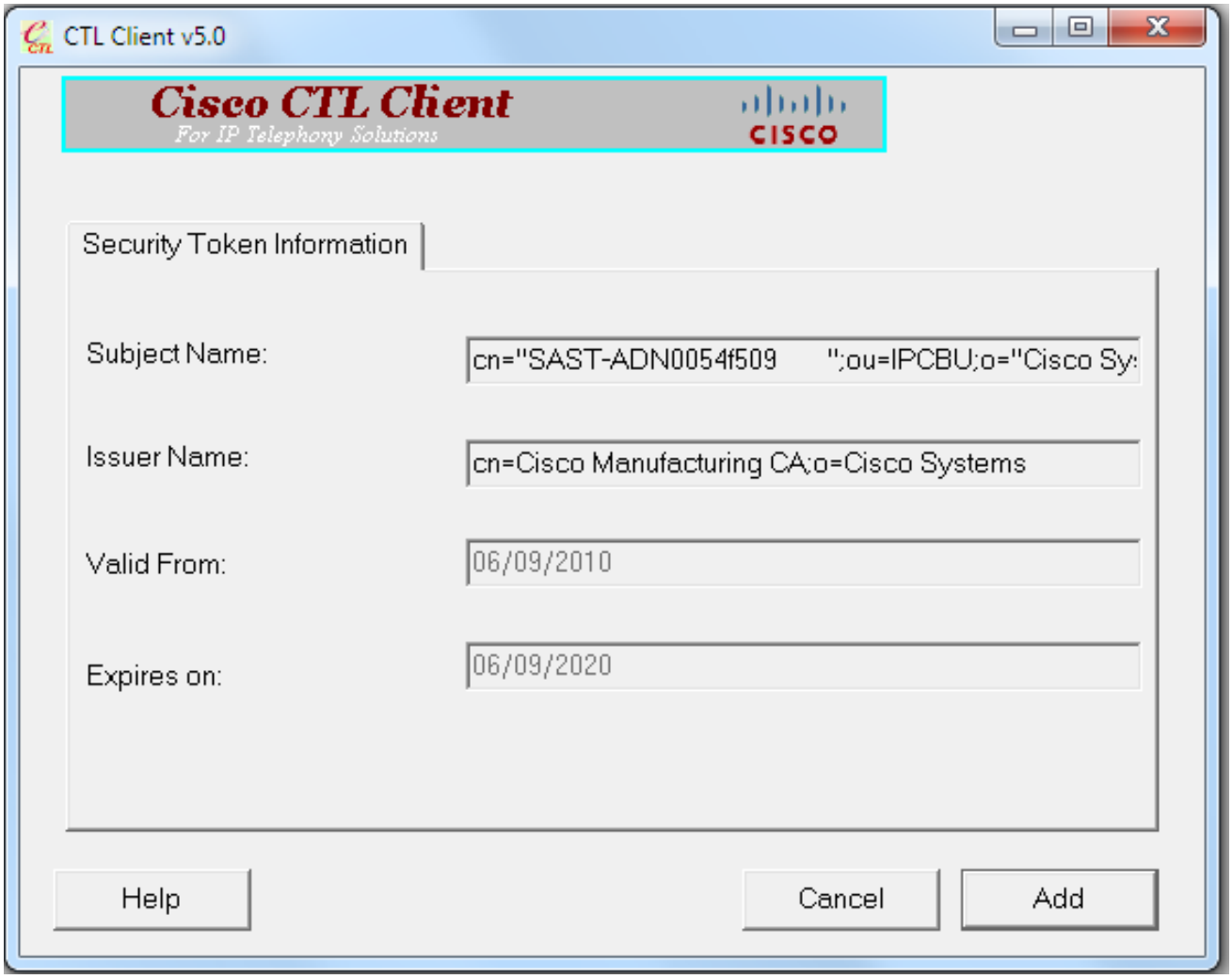
15. Depois que os detalhes do token de segurança forem exibidos, clique em **Add** (Adicionar):



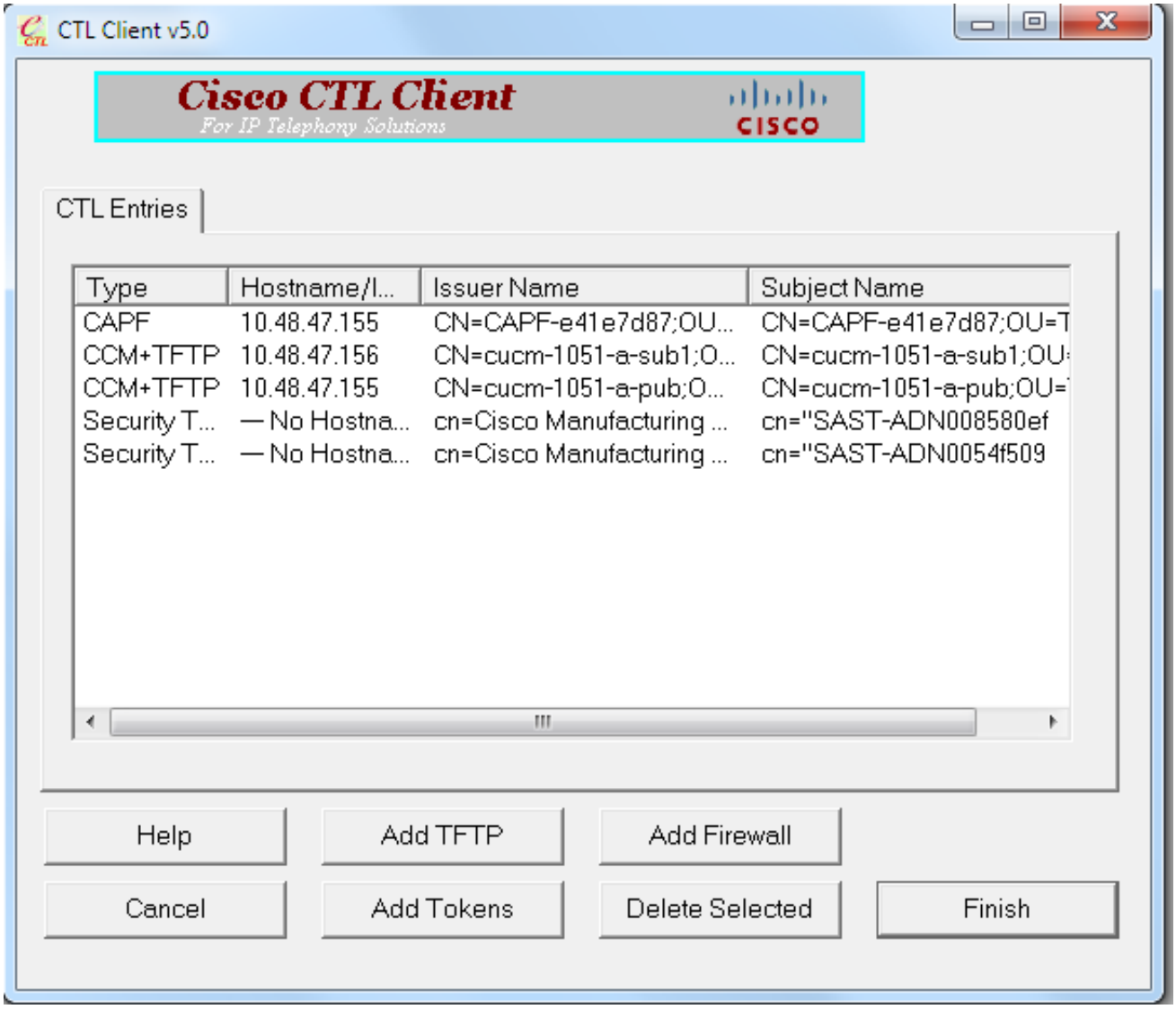
16. Quando o conteúdo do arquivo CTL aparecer, clique em **Add Tokens(Adicionar tokens) para adicionar o segundo eToken USB:**



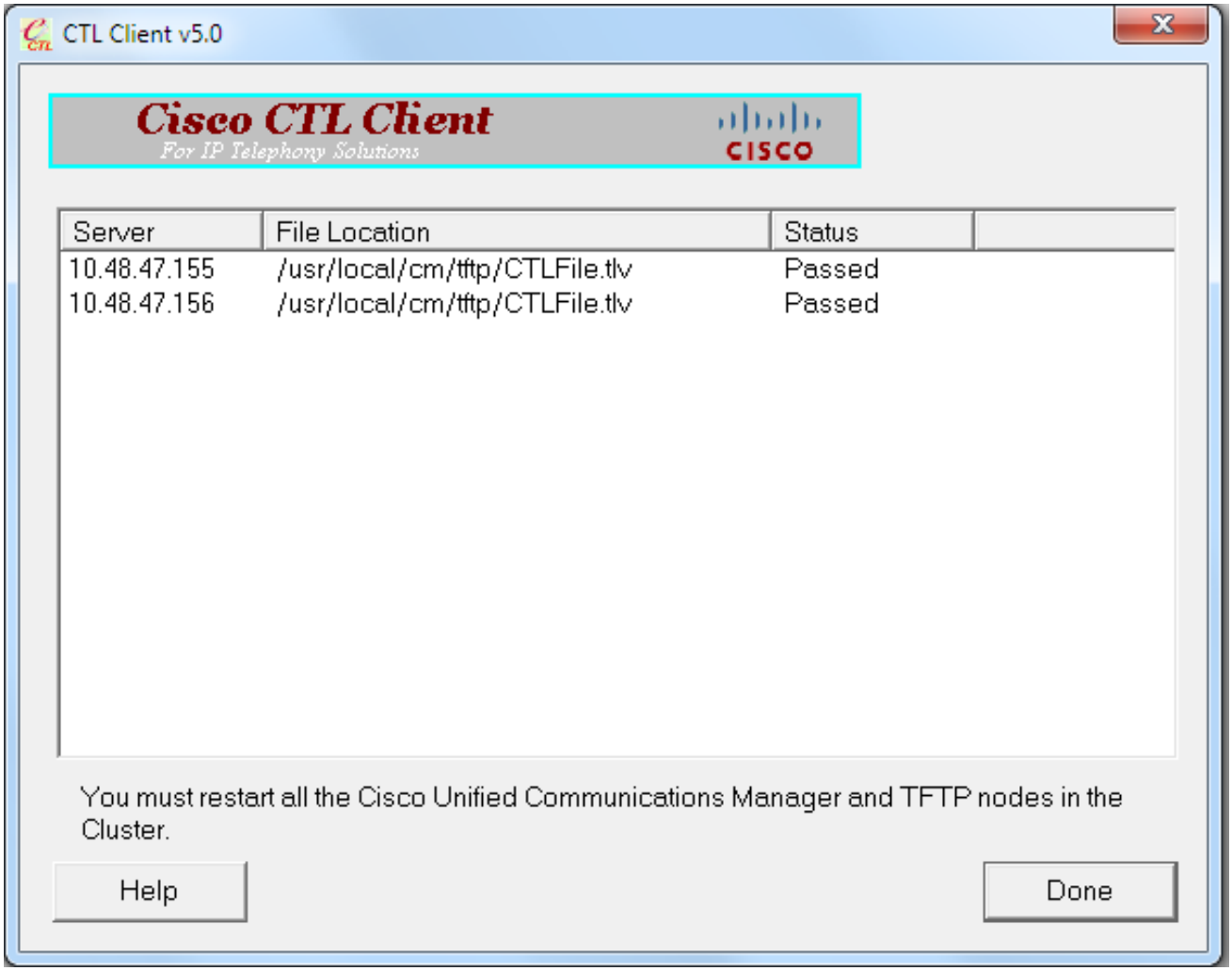
17. Depois que os detalhes do token de segurança forem exibidos, clique em **Add** (Adicionar):



18. Depois que o conteúdo do arquivo CTL for exibido, clique em **Finish**(Concluir). Quando solicitada uma senha, insira **Cisco123**:



19. Quando a lista de servidores de CUCM que contém a lista de arquivos aparecer, clique em **Done** (Concluído):



20. Reinicie os serviços TFTP e CallManager em todos os nós no cluster que os executam.

21. Reinicie todos os telefones IP para que eles possam obter a nova versão do arquivo CTL do serviço TFTP CUCM.

22. Para verificar o conteúdo do arquivo CTL, insira o comando **show ctl na CLI.** No arquivo CTL, você pode ver os certificados dos eTokens USB (um deles é usado para assinar o arquivo CTL). Veja 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 se depois de baixar os telefones IP, ele baixa a versão atualizada do arquivo CTL (as correspondências da soma de verificação MD5 são comparadas à saída do CUCM):

Essa alteração é possível porque você exportou e carregou anteriormente os certificados de eToken para o armazenamento confiável de certificado CUCM, e os telefones IP podem verificar esse certificado desconhecido usado para assinar o arquivo CTL com base no Trust Verification Service (TVS) executado no CUCM. Este trecho de registro ilustra como o telefone IP entra em contato com o TVS de CUCM com uma solicitação para verificar o certificado de eToken desconhecido, carregado como **Phone-SAST-Trust e é confiável:**

```
//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

Geração nova de certificado para solução CTL sem token

Esta seção descreve como recriar um certificado de segurança de cluster CUCM quando a solução CTL sem token é usada.

No processo de manutenção do CUCM, às vezes, o certificado do nó CallManager do editor do CUCM é alterado. Os cenários nos quais isso pode acontecer incluem a alteração do nome de host, a alteração do domínio ou apenas uma nova geração do certificado (devido à proximidade da data de vencimento do certificado).

Depois que o arquivo CTL é atualizado, ele é assinado com um certificado diferente do atual no arquivo CTL instalado em telefones IP. Normalmente, esse novo arquivo CTL não é aceito; no entanto, após o telefone IP encontrar o certificado desconhecido usado para assinar o arquivo CTL, ele entra em contato com o serviço TVS no CUCM.

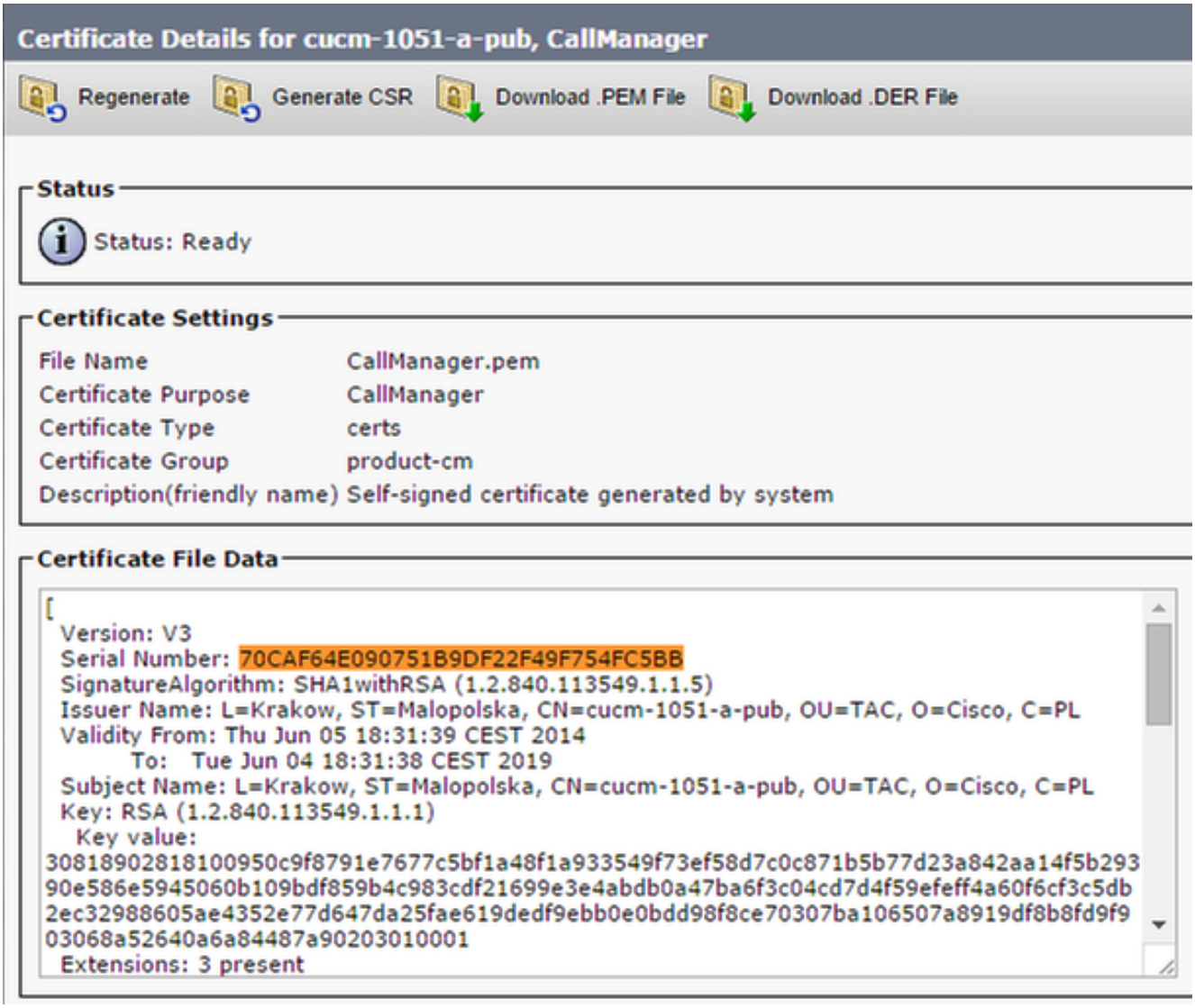
> **Note**: A lista de servidores em TVS está no arquivo de configuração de telefone IP e mapeada nos servidores de**Device Pool > CallManager Group** (Pool de dispositivos > Grupo

CallManager) do telefone IP.

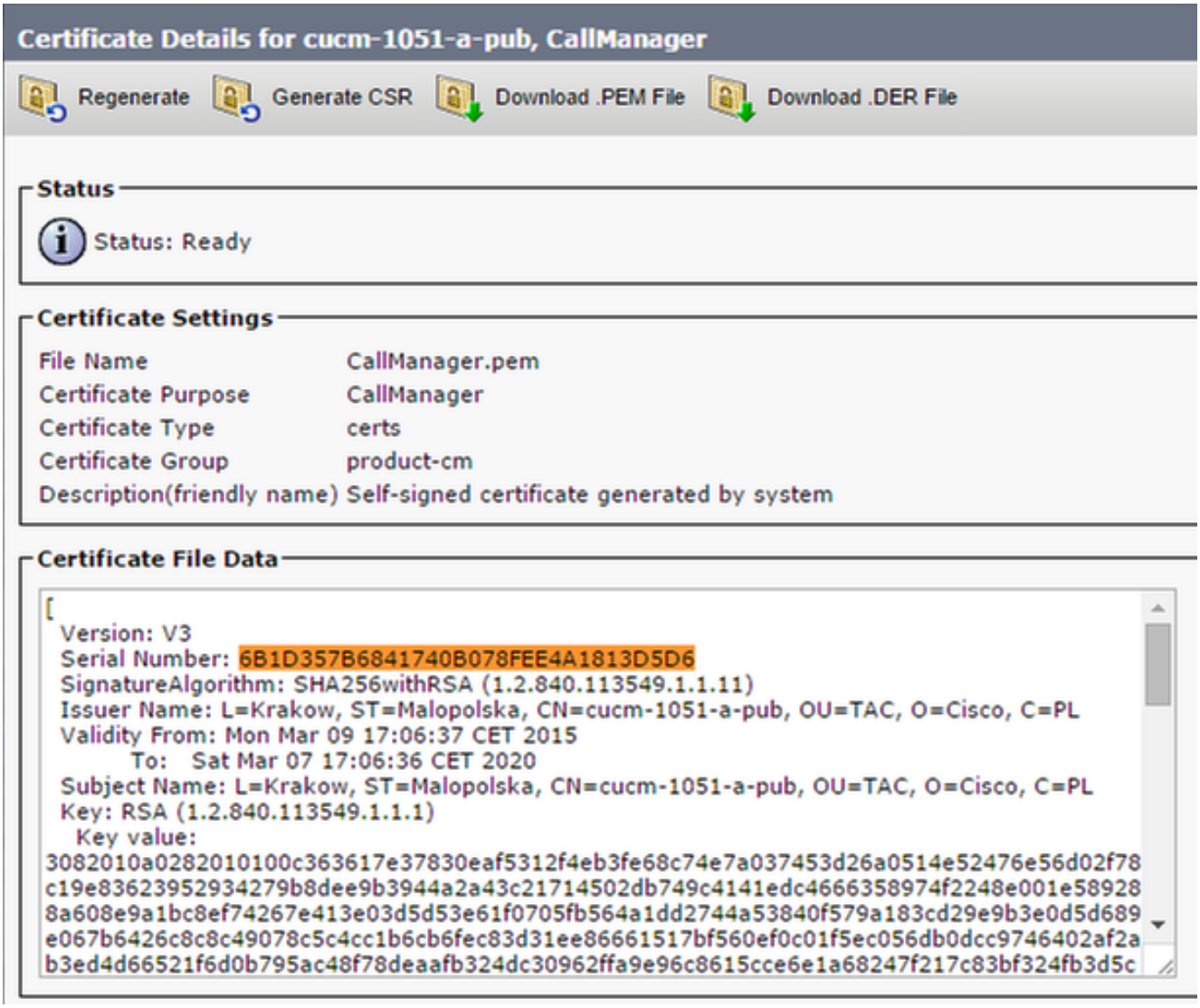
Após a verificação bem-sucedida no servidor de TVS, o telefone IP atualiza o arquivo CTL com a nova versão. Esses eventos ocorrem neste cenário:

1. O arquivo CTL existe no CUCM e no telefone IP. O certificado CCM+TFT (servidor) para o nó do editor CUCM é usado para 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 gerado novamente, e você pode ver o número de série das alterações de certificado:



3. O comando **utils ctl update CTLFile é inserido na CLI para 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 TVS atualiza o cache de certificado com os detalhes do novo 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. Ao visualizar o conteúdo do arquivo CTL, você pode ver que o arquivo está assinado com o novo certificado de servidor CallManager para o nó do Editor:

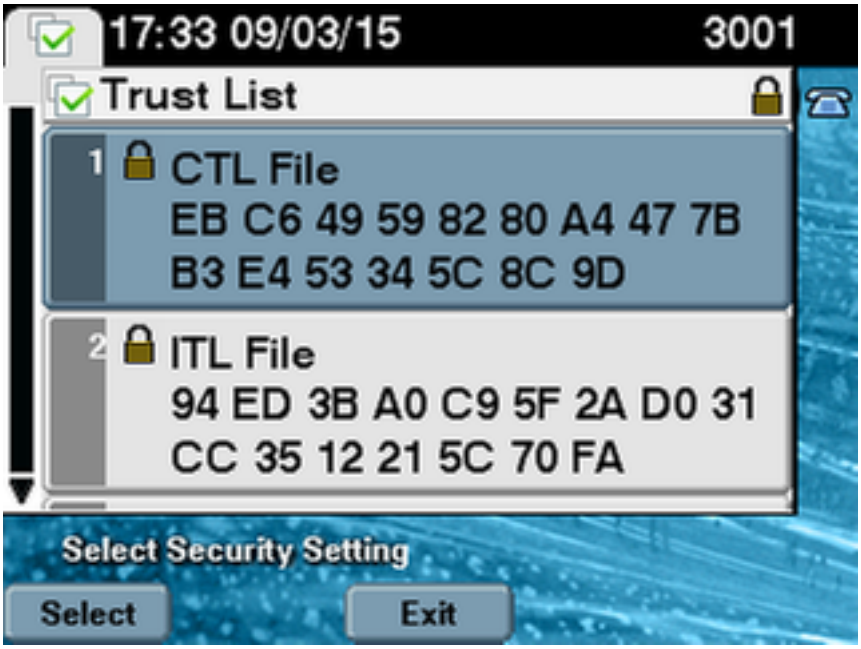
```
admin:show ctl
The checksum value of the CTL file:
ebc649598280a4477bb3e453345c8c9d(MD5)
ef5c006b6182cad66197fac6e6530f15d009319d(SHA1)

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. Na página Unified Serviceability (Unificação de manutenção) , os serviços TFTP e Cisco CallManager são reiniciados em todos os nós no cluster que os executam.

7. Os telefones IP são reiniciados e entram em contato com o servidor TVS para verificar o certificado desconhecido, que agora é usado para 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. Por fim, nos telefones IP, você pode verificar se o arquivo CTL é atualizado com a nova versão, e se a soma de verificação MD5 do novo arquivo CTL corresponde à do 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