Migrer CUCM en mode mixte avec CTL sans jeton

Table des matières

Introduction

Ce document décrit les différences entre la sécurité de Cisco Unified Communications Manager (CUCM) avec / sans l'utilisation d'eTokens USB matériels.

Conditions préalables

Exigences

Cisco vous recommande d'avoir des connaissances au sujet de CUCM version 10.0 (1) ou version ultérieure. En outre, vérifiez que :

- Votre serveur de licences pour CUCM version 11.5.1SU3 ou ultérieure doit être Cisco Prime License Manager (PLM) version 11.5.1SU2 ou ultérieure. Cela est nécessaire, car CUCM version 11.5.1SU3 exige que la licence de chiffrement active le mode mixte et PLM ne prend pas en charge la licence de chiffrement avant la version 11.5.1SU2.

Pour en savoir plus, consultez les notes de version pour Cisco Prime License Manager, version 11.5(1)SU2.

- Vous disposez d'un accès administratif à l'interface de ligne de commande (CLI) du nœud de l'éditeur CUCM.
- Vous avez accès aux eTokens USB du matériel et le module d'extension du client CTL est installé sur votre ordinateur pour les scénarios qui nécessitent une migration vers l'utilisation des jetons eTokens du matériel.

Pour plus de clarté, cette condition n'est requise que si vous disposez, à un moment donné, d'un scénario dans lequel les eTokens USB sont nécessaires. Les chances sont très faibles que les

eTokens USB sont nécessaires pour la plupart des gens.

- Il y a une connectivité complète entre tous les nœuds CUCM de la grappe. C'est très important, car le fichier CTL est copié sur tous les nœuds de la grappe via le protocole de transfert de fichier SSH (SFTP).
- La réplication de base de données (BD) dans la grappe fonctionne correctement et les serveurs répliquent les données en temps réel.
- Les appareils de votre déploiement prennent en charge la sécurité par défaut (TVS). Vous pouvez utiliser la liste des fonctionnalités de téléphone Unified CM de la page Web de <CUCM IP or FQDN>Cisco Unified Reporting (https:///cucreports/) afin de déterminer les appareils qui prennent en charge la sécurité par défaut.

Remarque : Cisco Jabber et de nombreux téléphones IP Cisco TelePresence ou Cisco 7940/7960 ne prennent pas actuellement en charge la sécurité par défaut. Si vous déployez la CTL sans jeton avec des périphériques qui ne prennent pas en charge la sécurité par défaut, toute mise à jour de votre système qui modifie le certificat CallManager sur l'éditeur empêche alors le fonctionnement normal de ces périphériques jusqu'à ce que la CTL soit supprimée manuellement. Les appareils qui prennent en charge la sécurité par défaut, tels que les téléphones 7945 et 7965 ou de versions plus récentes, peuvent installer des fichiers de CTL lorsque le certificat CallManager sur l'éditeur est mis à jour parce qu'ils peuvent utiliser le service de vérification de confiance (TVS).

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- CUCM version 10.5.1.10000-7 (grappe de deux nœuds)
- Les téléphones IP de la gamme Cisco 7975 qui ont été enregistrés via le protocole SCCP (Skinny Client Control Protocol) avec la version du micrologiciel SCCP75.9-3-1SR4-1S
- Deux jetons de sécurité Cisco utilisés pour définir la grappe en mode mixte avec l'utilisation du logiciel client CTL

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

Ce document décrit la différence sur le plan de la sécurité de Cisco Unified Communication Manager (CUCM) avec et sans les jetons eTokens USB matériels.

Ce document décrit également les scénarios de mise en œuvre de base qui font appel à la liste de certificats de confiance sans jetons (CTL) et le processus utilisé pour veiller à ce que le système fonctionne correctement après les modifications.

La CTL sans jeton est une nouvelle fonctionnalité de CUCM version 10.0(1) ou ultérieure qui permet le chiffrement de la signalisation des appels et des supports pour les téléphones IP sans avoir à utiliser de jetons eTokens USG matériels et de module d'extension de client CTL, qui constituaient des exigences dans les versions CUCM précédentes.

Lorsque la grappe est placée en mode mixte à l'aide de la commande CLI, le fichier CTL est signé avec le certificat de CCM et de TFTP du nœud de l'éditeur et aucun certificat eToken n'est présent dans le fichier CTL.

Remarque : lorsque vous régénérez le certificat CallManager (CCM+TFTP) sur l'éditeur, il modifie le signataire du fichier. Les téléphones et les périphériques qui ne prennent pas en charge la sécurité par défaut n'acceptent pas non plus le nouveau fichier CTL, sauf si les fichiers CTL sont supprimés manuellement de chaque périphérique. Pour en savoir plus, reportez-vous à la dernière exigence énumérée dans la section des Conditions préalables du présent document.

Du mode non sécurisé au mode mixte (CTL sans jetons)

Cette section décrit le processus utilisé pour transférer la sécurité de la grappe CUCM en mode mixte via l'interface de ligne de commande.

Avant ce scénario, le CUCM était en mode non sécurisé, ce qui signifie qu'il n'y avait pas de fichier CTL présent sur l'un des nœuds et que seul le fichier ITL (Identity Trust List) avait été installé sur les téléphones IP enregistrés, comme illustré dans les résultats suivants :

<#root>

```
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:
```

Remarque : si un fichier CTL a été trouvé sur le serveur alors que le cluster n'est pas en mode mixte, cela signifie que le cluster a été une fois en mode mixte, puis est revenu en mode non mixte et que le fichier CTL n'a pas été supprimé du cluster.

La commande file delete activelog cm/tftpdata/CTLFile.tlv supprime le fichier CTL des noeuds du cluster CUCM ; cependant, la commande doit être entrée sur chaque noeud. Autrement dit, utilisez cette commande uniquement si vos serveurs disposent d'un fichier

---

CTL et si la grappe n’est pas en mode mixte.

Pour vérifier facilement si une grappe est en mode mixte, utilisez la commande run sql select paramname,paramvalue from processconfig where paramname=’ClusterSecurityMode'. Si la valeur du paramètre est 0, la grappe n’est pas en mode mixte.

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

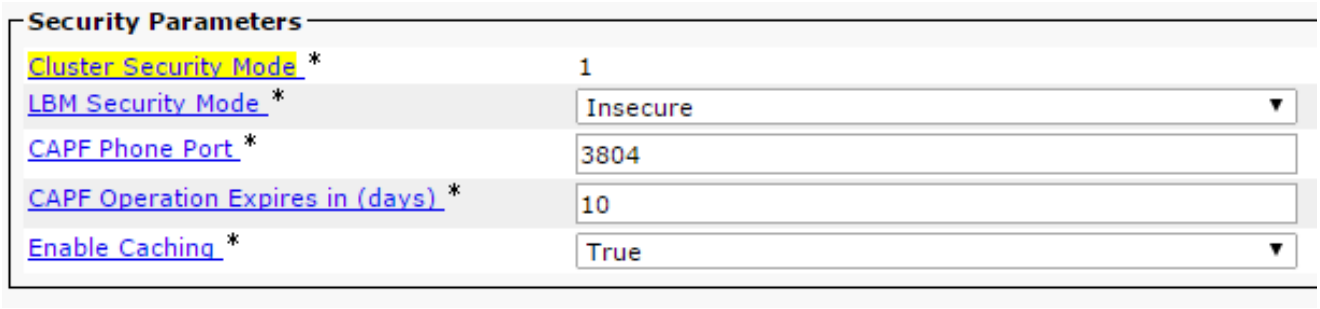
Pour faire passer la sécurité de la grappe CUCM en mode mixte à l’aide de la nouvelle fonctionnalité CTL sans jetons, procédez comme suit :

1. Obtenez l’accès administratif à l’interface de ligne de commande du nœud de l’éditeur CUCM.
2. Saisissez la commande utils ctl set-cluster mixed-mode dans l’interface de ligne de commande :

<#root>

```
admin:

utils ctl set-cluster mixed-mode


This operation sets 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. Accédez à CUCM Admin Page > System > Enterprise Parameters pour consulter les paramètres d'entreprise et vérifier si la grappe a été définie en mode mixte (une valeur de 1 indique le mode mixte) :



4. Redémarrez les services TFTP et Cisco CallManager sur tous les nœuds dans la grappe qui exécutent ces services.
5. Redémarrez tous les téléphones IP afin qu'ils puissent obtenir le fichier CTL à partir du serveur TFTP de CUCM.
6. Afin de vérifier le contenu du fichier CTL, saisissez la commande show ctl dans l'interface de ligne de commande.
7. Dans le fichier de liste CTL, vous pouvez voir que le certificat CCM + TFTP (serveur) pour le nœud de l'éditeur CUCM est utilisé pour signer le fichier de la CTL (ce fichier est le même sur tous les serveurs de la grappe). Voici un exemple de sortie :

<#root>

```
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
```

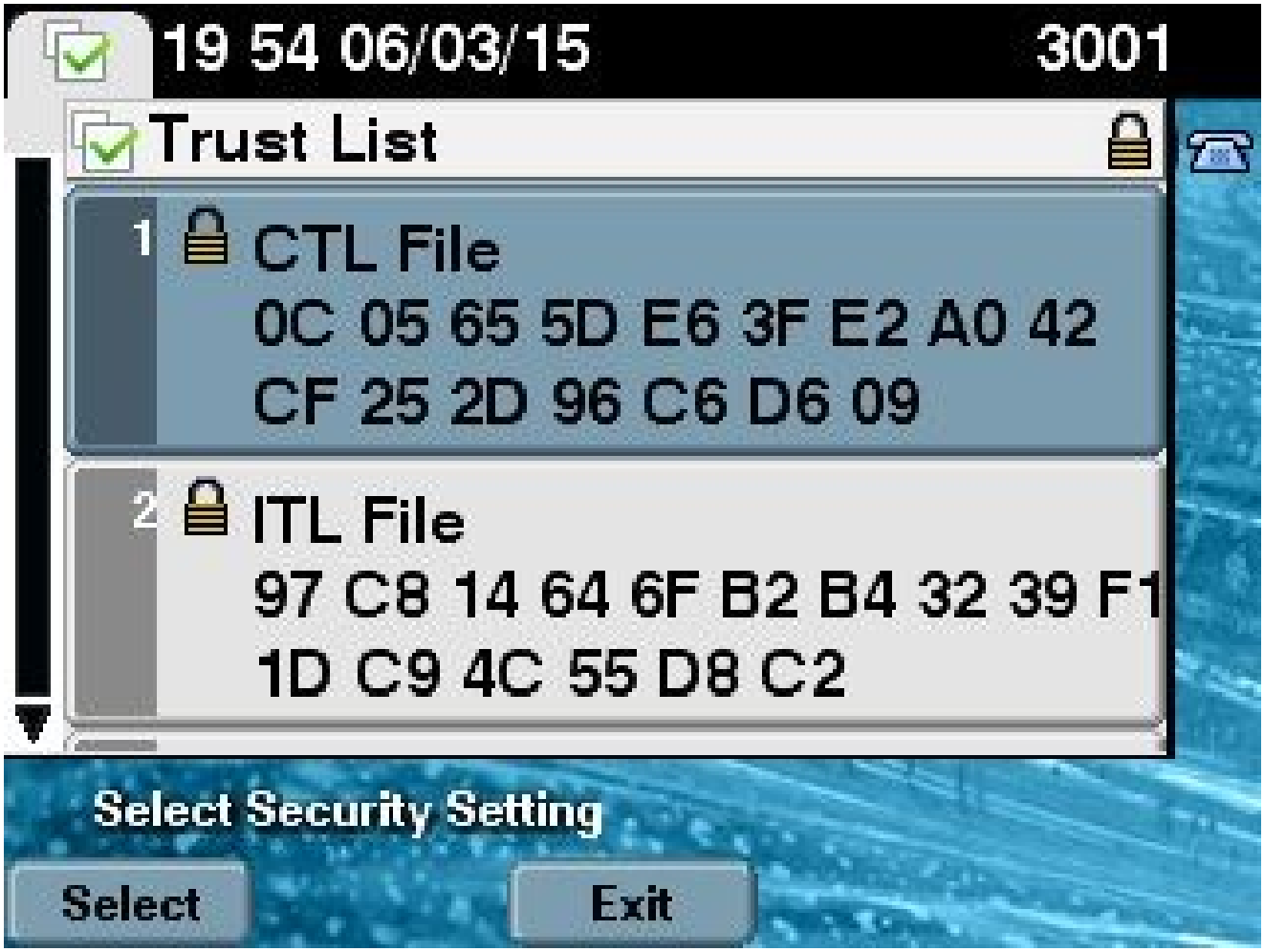
70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB

```
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.
```

8. Sur le téléphone IP, vous pouvez vérifier qu'après le redémarrage du service, il télécharge le fichier CTL, qui est maintenant présent sur le serveur TFTP (la somme de contrôle MD5 correspond lorsqu'elle est comparée au résultat de CUCM) :

> Remarque : lorsque vous vérifiez la somme de contrôle sur le téléphone, vous voyez MD5 ou SHA1, selon le type de téléphone.



Des jetons eTokens matériels à la solution sans jetons

Cette section décrit comment faire migrer la sécurité de grappe CUCM des jetons eTokens matériels à l'utilisation de la nouvelle solution sans jetons.

Dans certaines situations, le mode mixte est déjà configuré sur le CUCM à l'aide du client CTL et les téléphones IP utilisent les fichiers CTL qui contiennent les certificats des jetons eTokens matériels.

Dans ce scénario, le fichier CTL est signé par un certificat à partir de l'un des jetons eTokens USB et est installé sur les téléphones IP. Par exemple :

<#root>

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

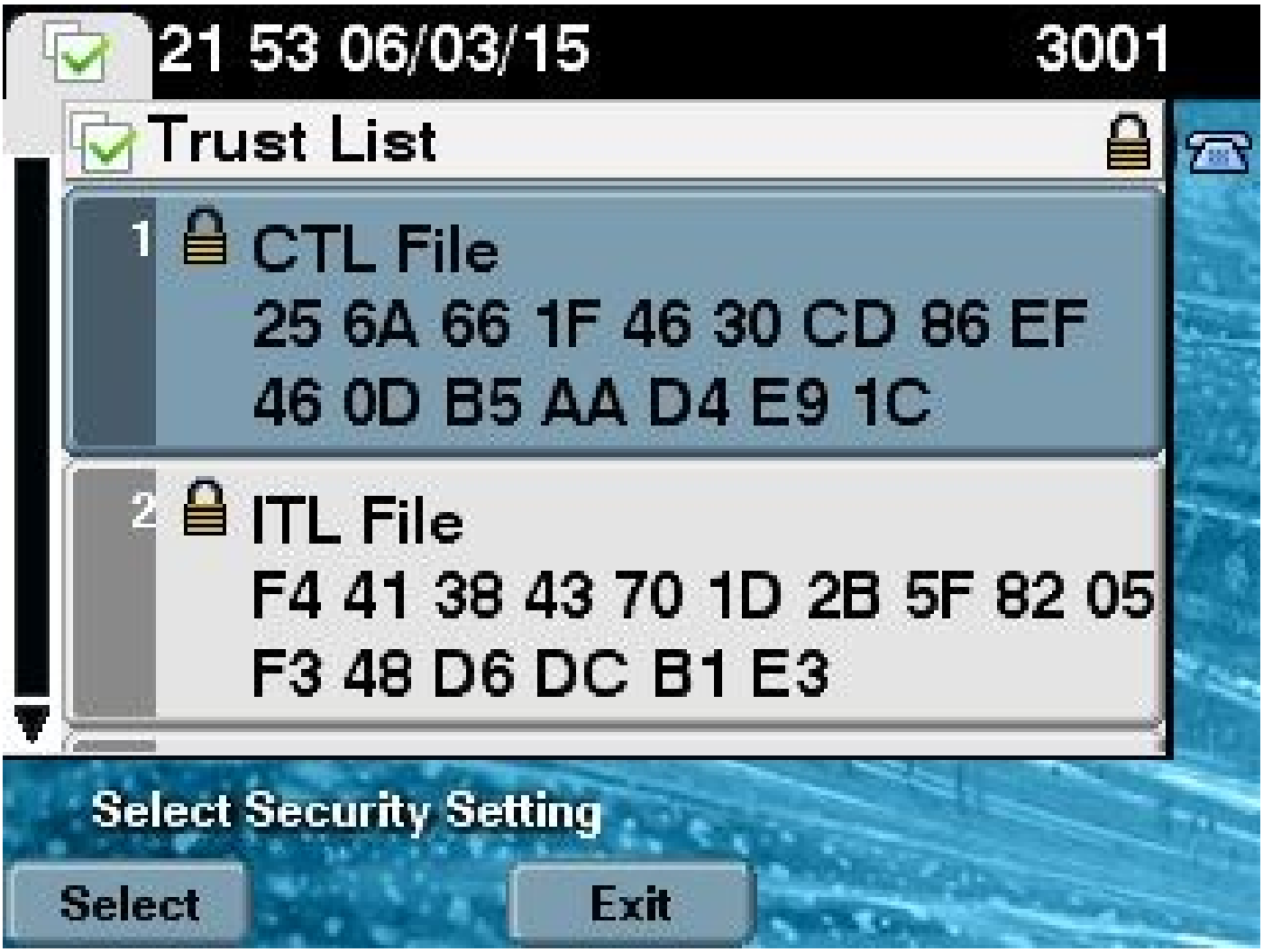
BYTEPOS	TAG	LENGTH	VALUE
1	RECORDLENGTH	2	1186
2	DNSNAME	1	
3	SUBJECTNAME	56	cn="SAST-ADN00858Oef ";ou=IPCBU;o="Cisco Systems
4	FUNCTION	2	System Administrator Security Token
5	ISSUERNAME	42	cn=Cisco Manufacturing CA;o=Cisco Systems
6	SERIALNUMBER	10	

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.



Procédez comme suit pour faire un virage de la sécurité de grappe CUCM à l'utilisation de CTL sans jetons :

1. Obtenez l'accès administratif à l'interface de ligne de commande du nœud de l'éditeur CUCM.
2. Saisissez la commande CLI utils ctl update CTLFile :

```
<#root>

admin:

utils ctl update CTLFile


This operation updates 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. Redémarrez les services TFTP et CallManager sur tous les nœuds dans la grappe qui exécutent ces services.

4. Redémarrez tous les téléphones IP afin qu'ils puissent obtenir le fichier CTL à partir du serveur TFTP de CUCM.
5. Saisissez la commande show ctl dans l'interface de ligne de commande afin de vérifier le contenu du fichier CTL. Dans le fichier de liste CTL, vous pouvez voir que le certificat CCM + TFTP (serveur) pour le nœud de l'éditeur CUCM est utilisé pour signer le fichier de la CTL au lieu du certificat des jetons eTokens USB matériels.
6. Une différence plus importante dans ce cas est que les certificats de tous les jetons eTokens USB matériels sont supprimés du fichier CTL. Voici un exemple de sortie :

<#root>

```
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.
```

Remarque : dans le résultat, si le certificat CCM+TFTP (serveur) du serveur de publication CUCM n'est pas signataire, revenez au mode de sécurité de cluster basé sur le jeton matériel et répétez les modifications à nouveau pour la solution sans jeton.

7. Pour ce qui concerne les téléphones IP, vous pouvez vérifier qu’après le redémarrage des téléphones IP, la version du fichier CTL mis à jour a été téléchargée (la somme de contrôle MD5 correspond lorsqu’elle est comparée au résultat de CUCM) :

De la solution sans jetons aux jetons eTokens matériels

Cette section décrit comment faire migrer la sécurité de grappe CUCM en l'éloignant de la nouvelle solution sans jetons afin de recommencer à utiliser les jetons matériels eTokens.

Lorsque la sécurité de grappe est définie en mode mixte à l'aide des commandes CLI, le fichier CTL est signé avec le certificat de CCM et de TFTP (serveur) pour le nœud de l'éditeur CUCM et il n'y a aucun certificat eTokens USB matériels dans le fichier de CTL.

Par conséquent, lorsque vous exécutez le client CTL pour mettre à jour le fichier CTL (et revenir à l'utilisation des eTokens matériels), le message d'erreur suivant s'affiche :

```
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.
```

Cela est particulièrement important dans les scénarios qui comportent un déclassement du système (lorsqu'une version précédente est rétablie) vers une version antérieure à 10.x qui n'inclut pas les commandes utils ctl .

Le fichier CTL précédent est migré (sans modification de son contenu) dans le cadre du processus d'actualisation ou de mise à niveau Linux à Linux (de couche 2), et il ne contient pas les certificats eToken, comme indiqué précédemment. Voici un exemple de sortie :

<#root>

```
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:
```

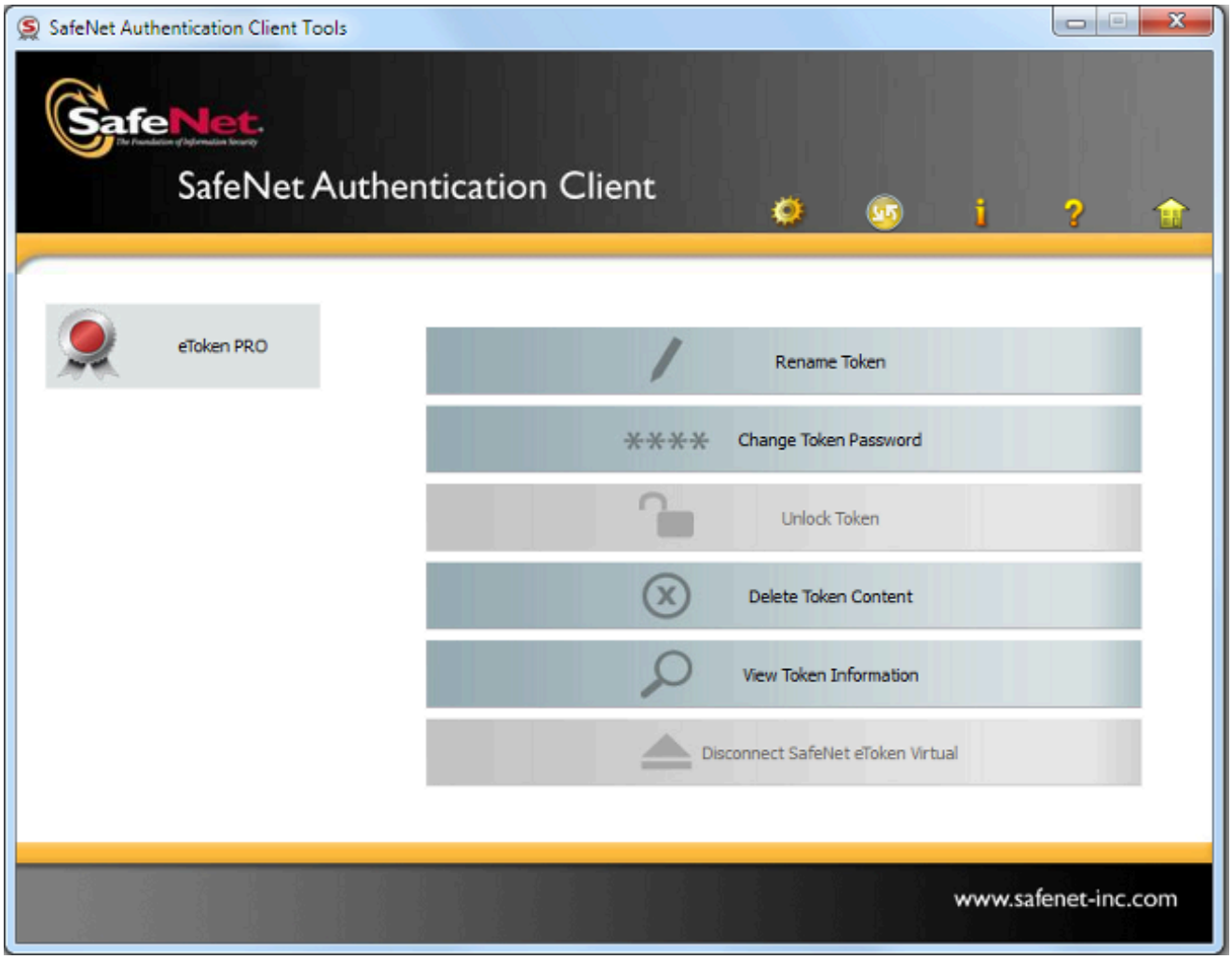
Pour ce scénario, procédez comme suit afin de mettre à jour de manière sécurisée les fichiers de la CTL sans avoir à utiliser la procédure pour les eTokens perdus, qui se termine par la suppression manuelle du fichier CTL de tous les téléphones IP :

1. Obtenez l'accès administratif à l'interface de ligne de commande du nœud de l'éditeur CUCM.
2. Saisissez la commande file delete tftp CTLFile.tlv dans l'interface de ligne de commande du nœud de l'éditeur afin de supprimer le fichier CTL :

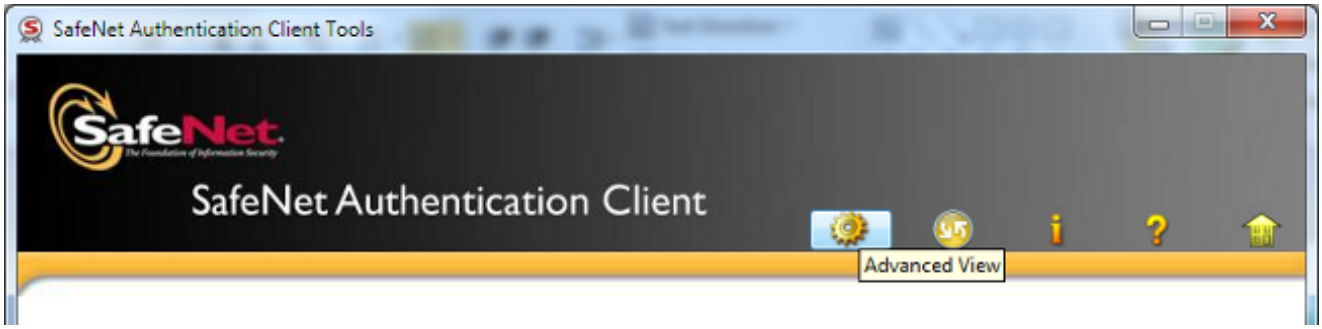
<#root>

```
admin:

file delete tftp CTLFile.tlv


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

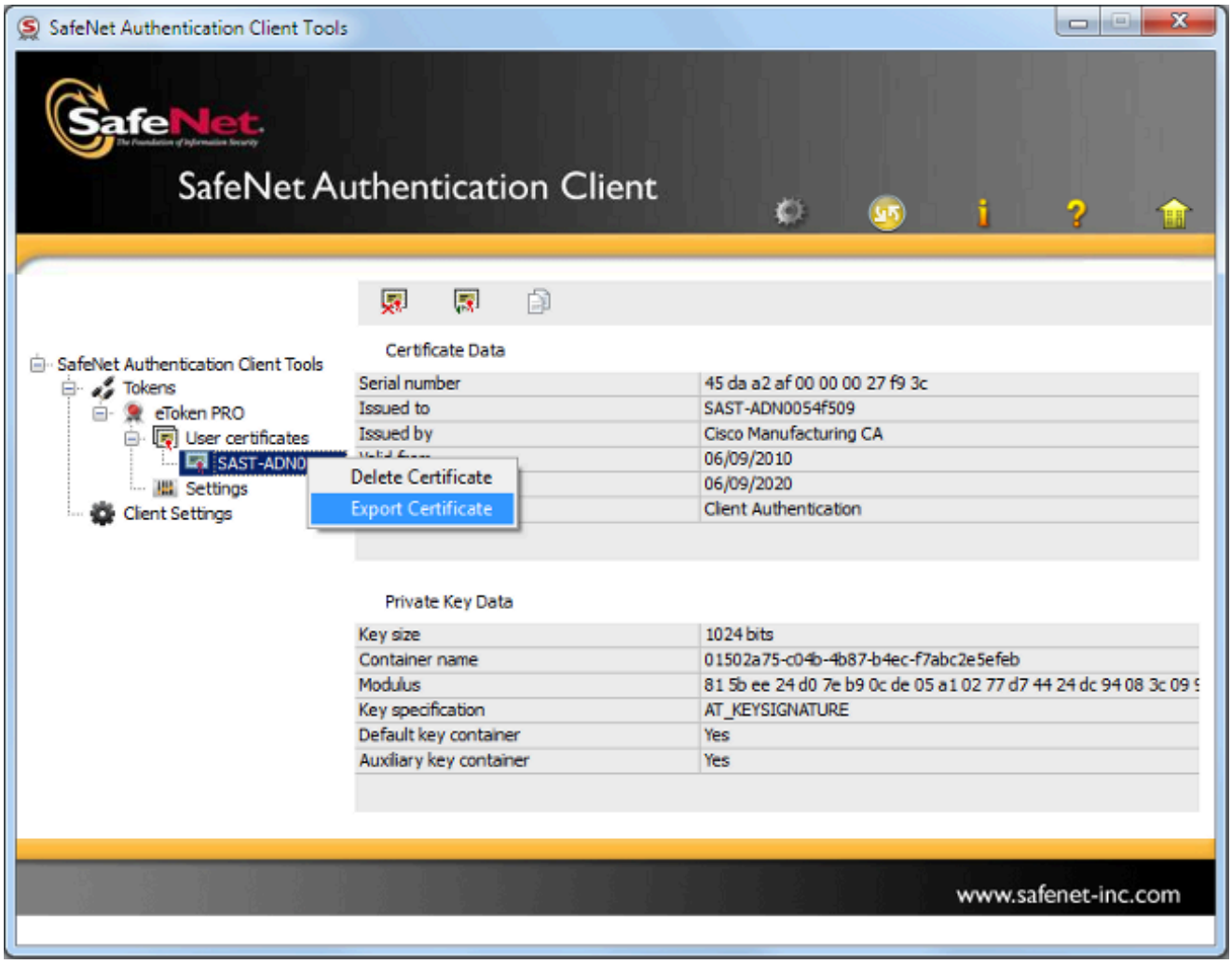
3. Ouvrez le client d'authentification SafeNet sur l'ordinateur Microsoft Windows sur lequel le client CTL est installé (il est automatiquement installé avec le client CTL) :



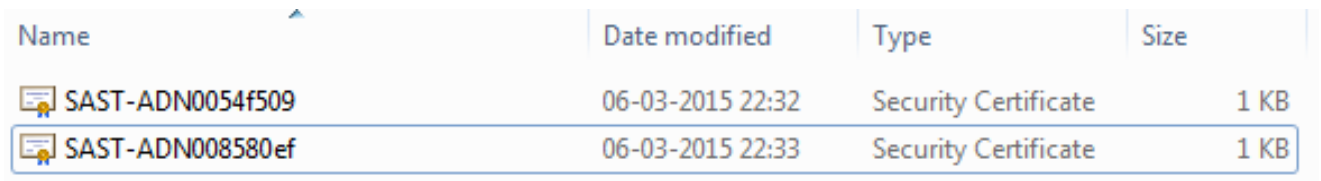
4. Dans le client d’authentification SafeNet, accédez à Advanced View (affichage avancé): :



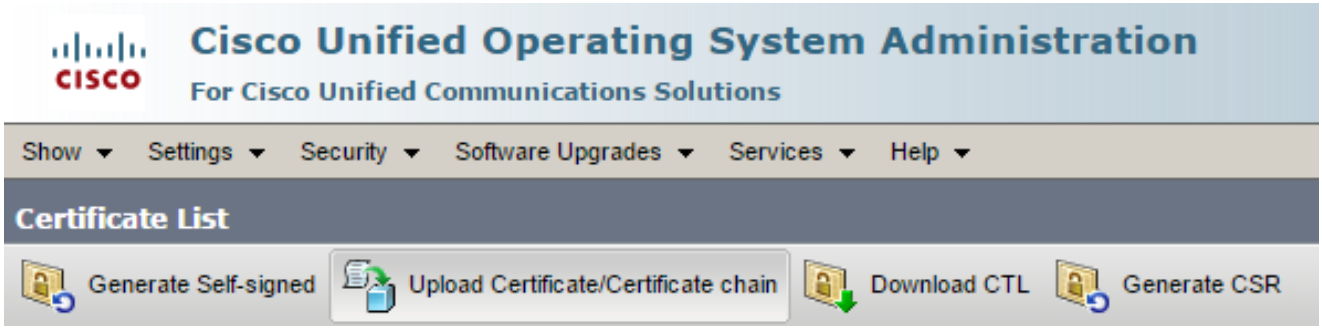
5. Insérez le premier eoken USB matériel.
6. Sélectionnez le certificat dans le dossier User certificates (certificats de l’utilisateur) et exportez-le dans le dossier de l’ordinateur. Lorsque vous êtes invité à saisir un mot de passe, utilisez le mot de passe par défaut Cisco123 :



7. Répétez ces étapes pour le deuxième eToken USB matériel afin que les deux certificats soient exportés vers l'ordinateur :

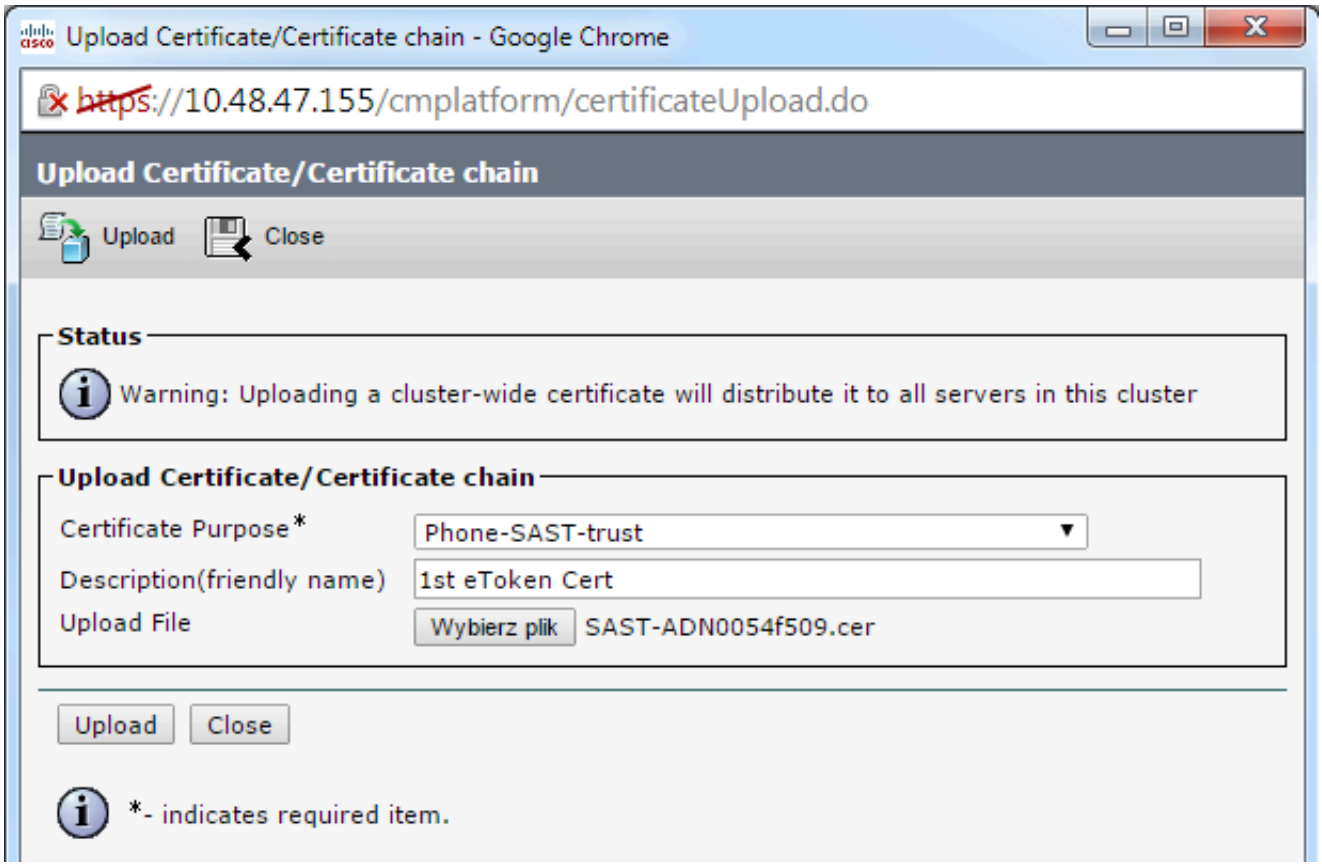


8. Connectez-vous à l'administration du système d'exploitation de Cisco Unified et accédez à Security > Certificate Management > Upload Certificate pour charger les certificats :

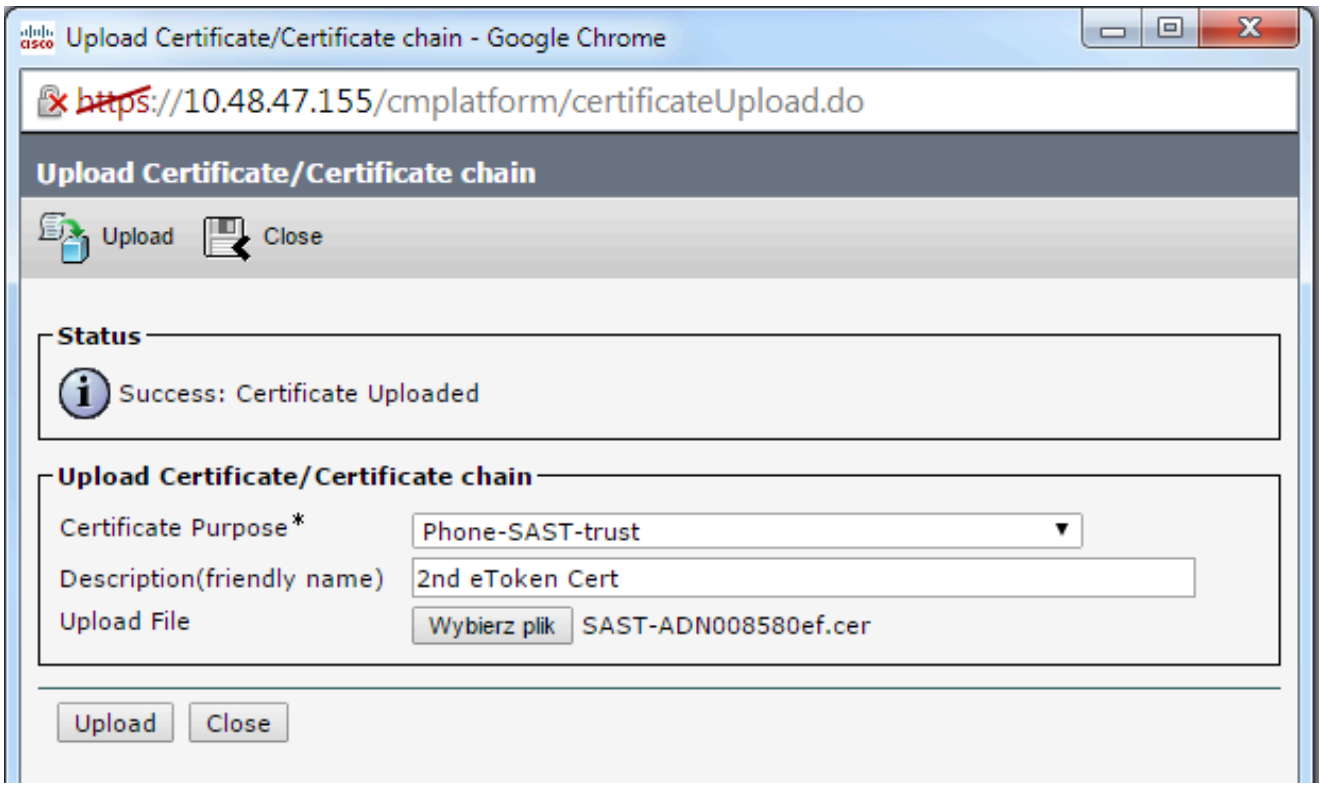


9. La page Upload Certificate (charger le certificat) s'affiche. Sélectionnez Phone-SAST-trust

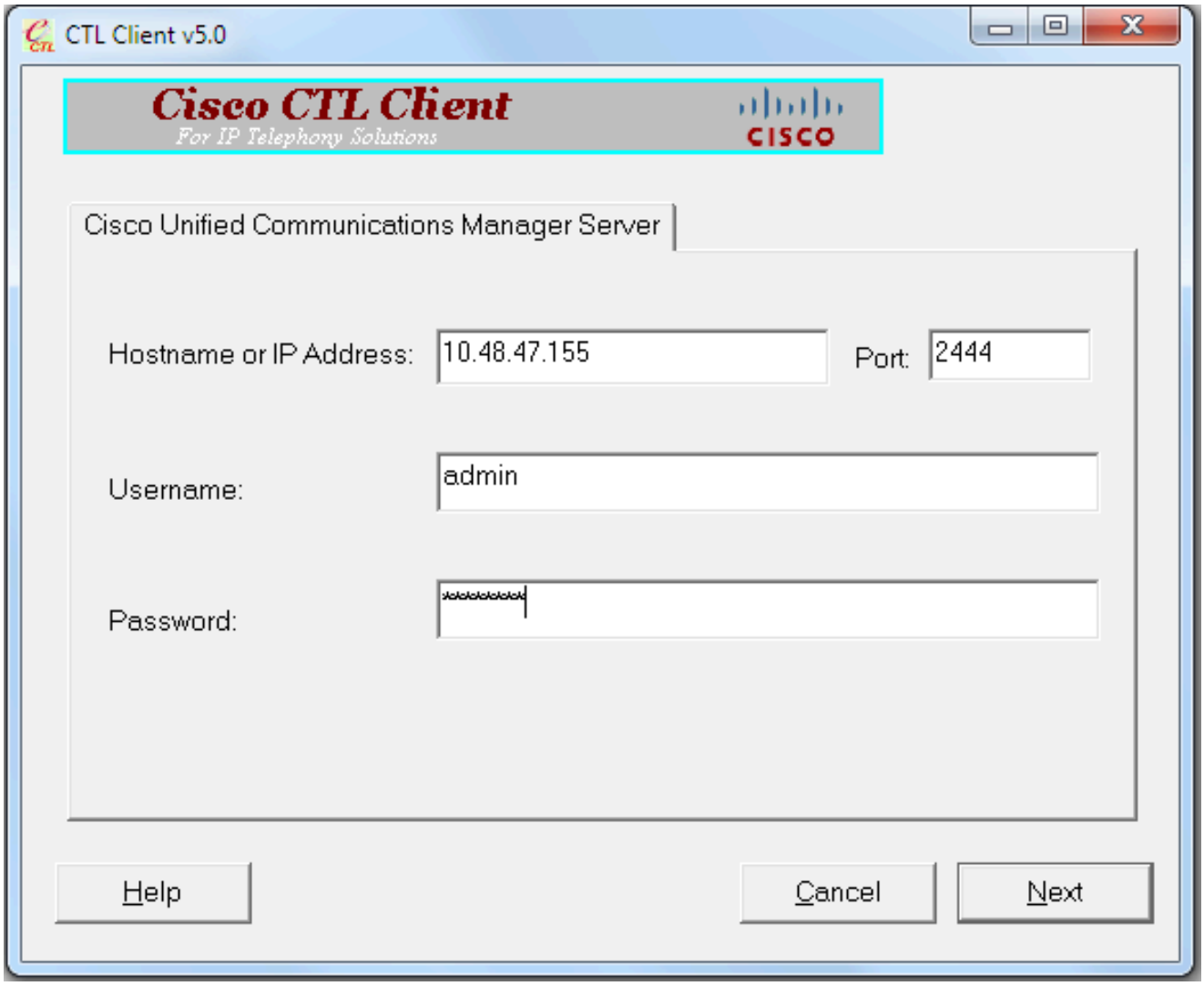
dans le menu déroulant de l'objet du certificat (Certificate Purpose) et sélectionnez le certificat que vous avez exporté du premier eToken :



10. Effectuez les étapes précédentes afin de charger le certificat que vous avez exporté à partir du deuxième eToken :



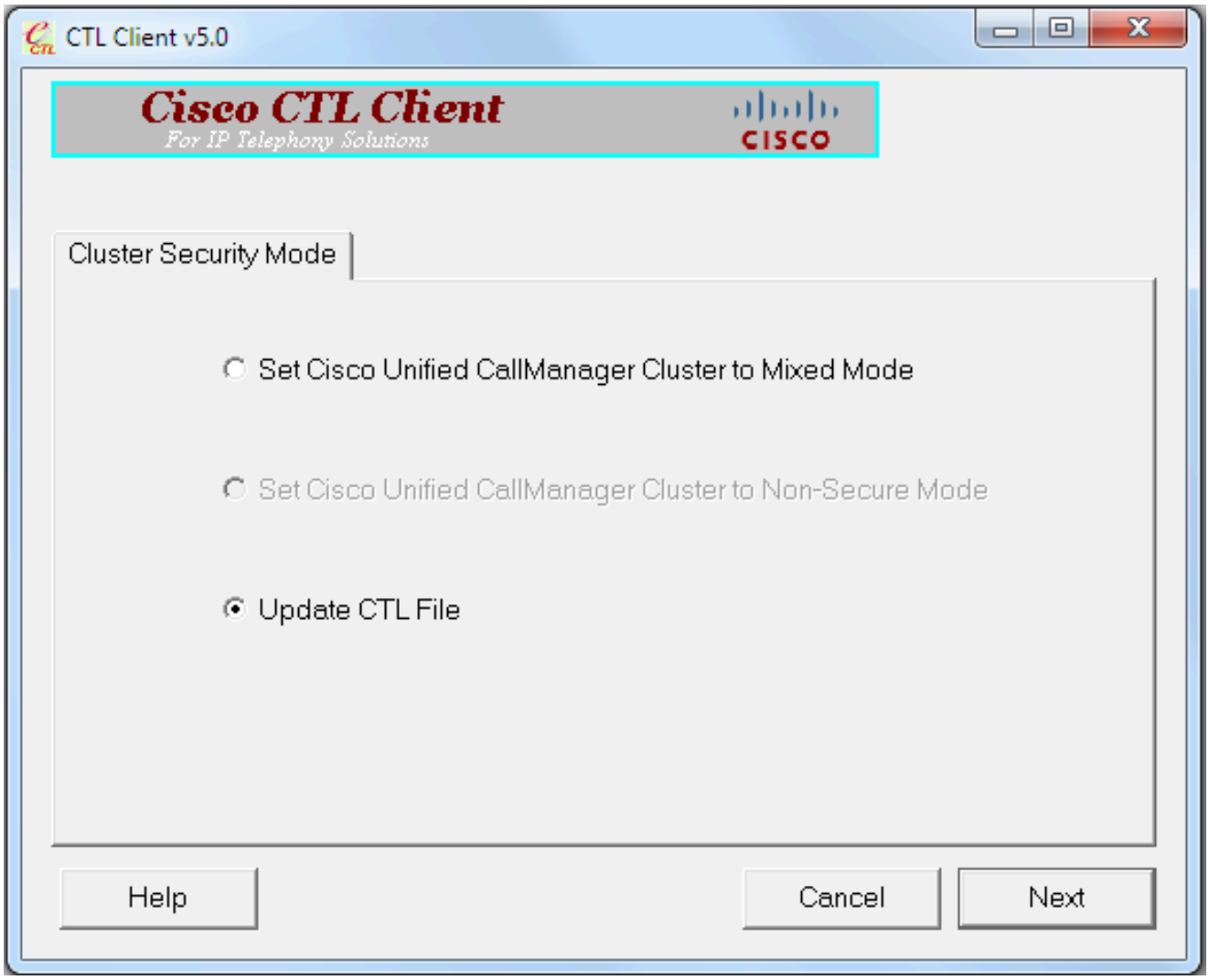
11. Exécutez le client CTL, indiquez l'adresse IP/le nom d'hôte du nœud d'éditeur CUCM, puis saisissez les informations d'identification de l'administrateur CCM :



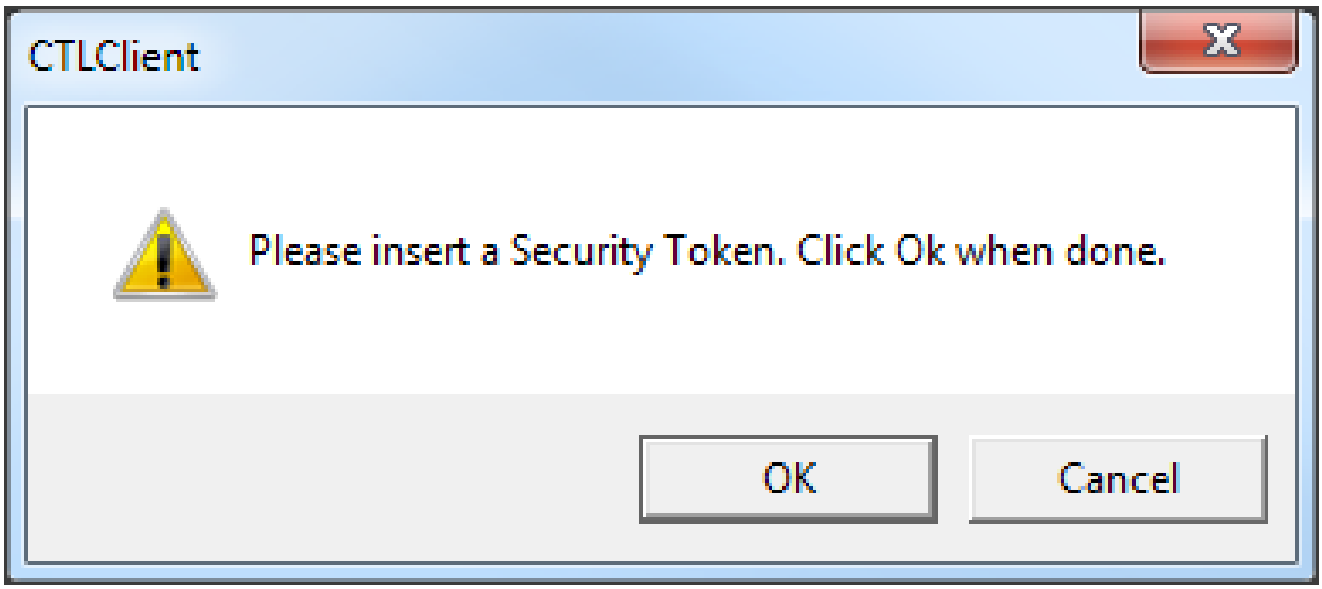
12. Étant donné que la grappe est déjà en mode mixte, mais qu'aucun fichier CTL n'existe sur le nœud de l'éditeur, ce message d'avertissement s'affiche (cliquez sur OK pour l'ignorer) :

```
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.
```

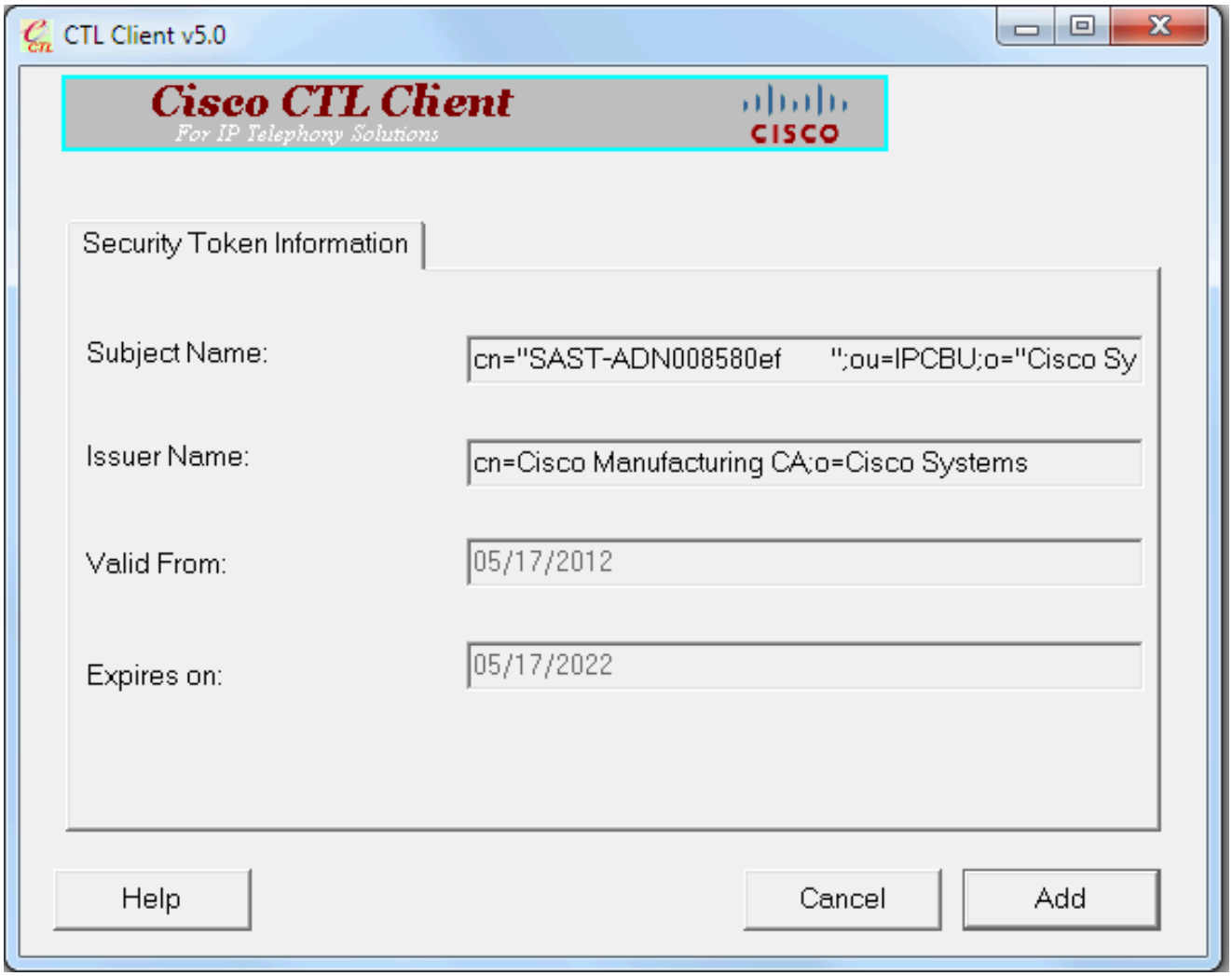
13. Dans le client CTL, cliquez sur la case d'option Update CTL File, puis cliquez sur Next :



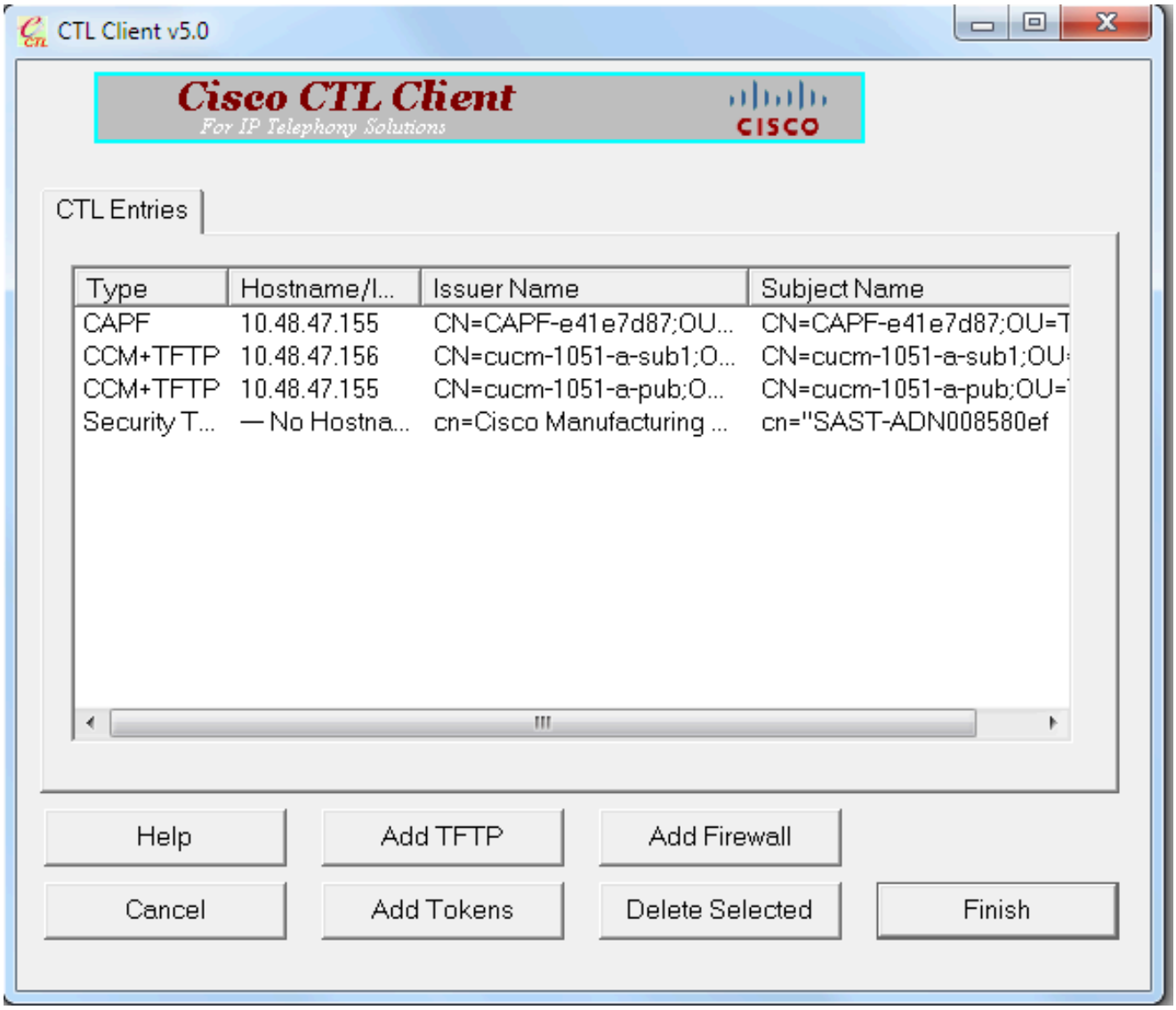
14. Insérez le premier jeton de sécurité et cliquez sur OK :



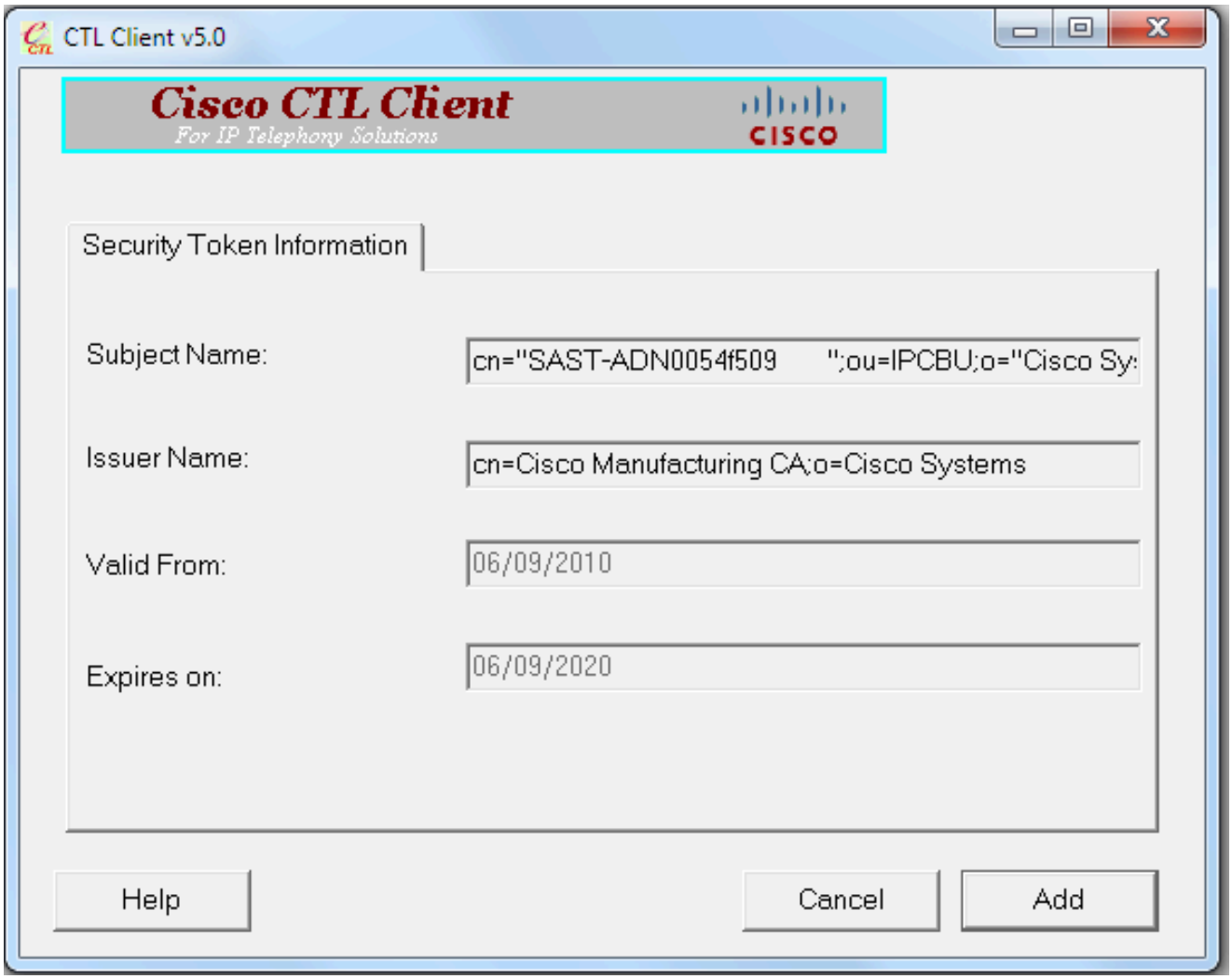
15. Une fois les informations relatives au jeton de sécurité affichées, cliquez sur Add (ajouter) :



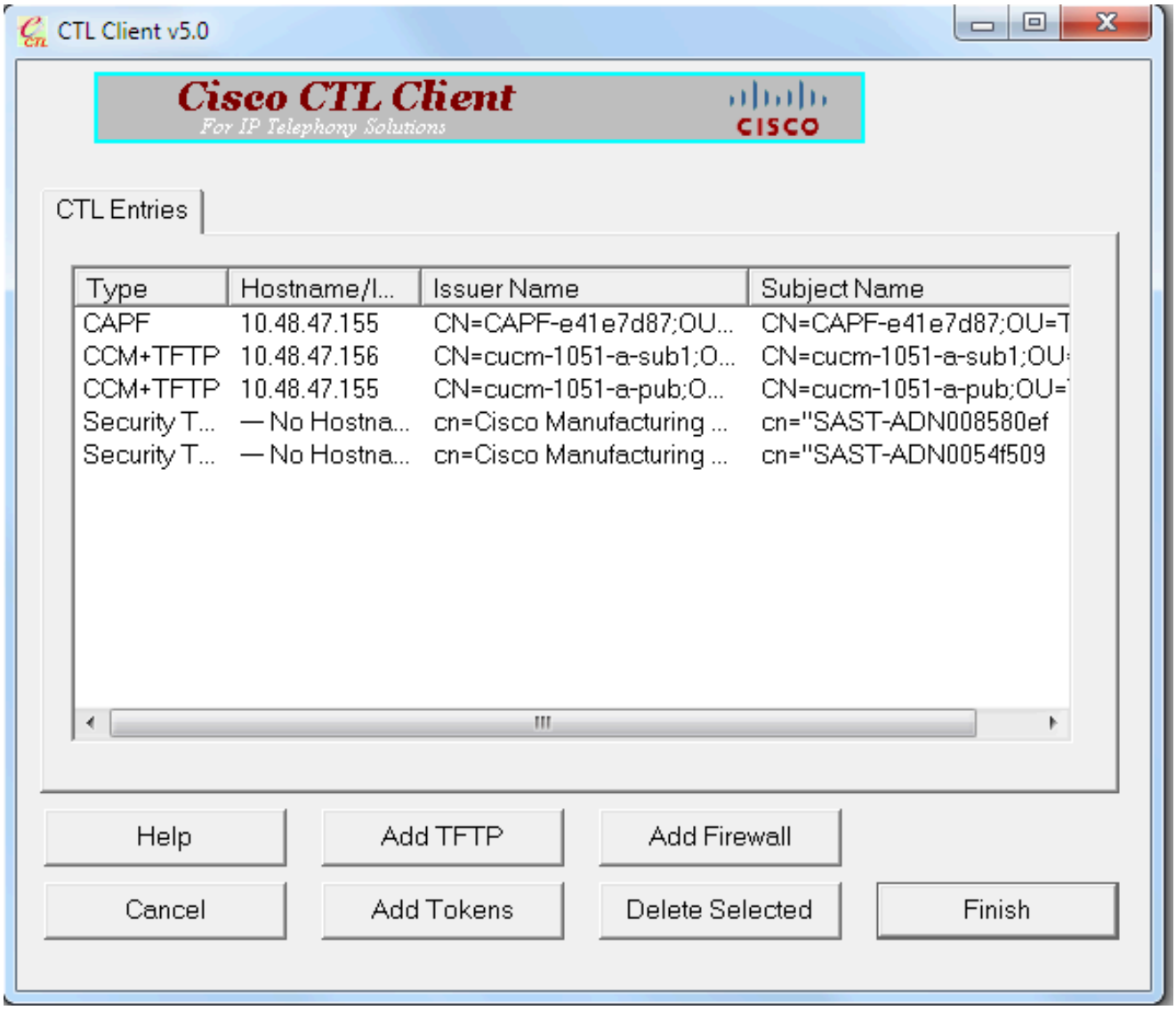
16. Une fois le contenu du fichier CTL affiché, cliquez sur Add Tokens (ajouter des jetons) afin d'ajouter le deuxième jeton eToken USB :



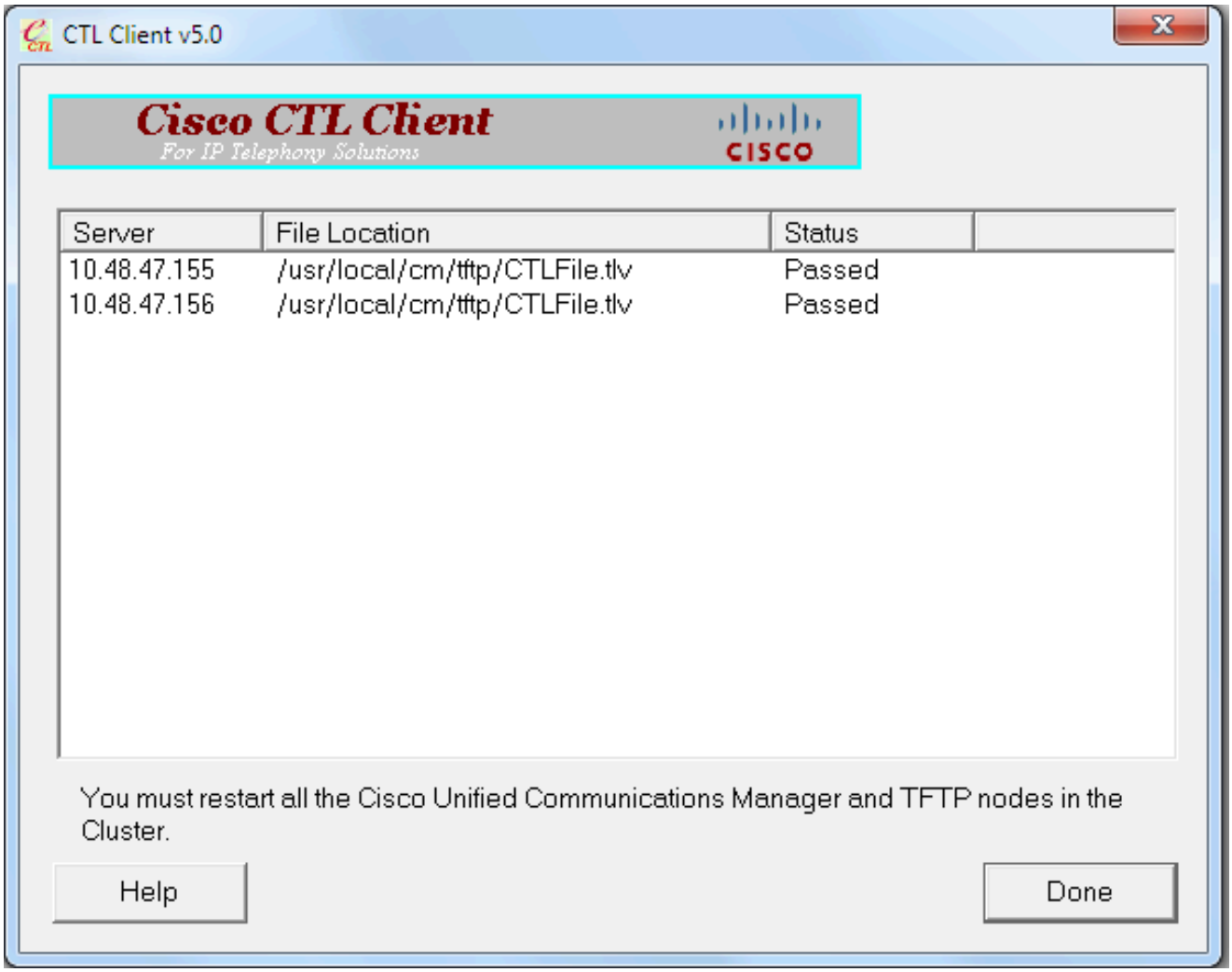
17. Lorsque les informations relatives au jeton de sécurité apparaissent, cliquez sur Add (ajouter) :



18. Une fois le contenu du fichier CTL affiché, cliquez sur Finish (teminer). Lorsque vous êtes invité à saisir un mot de passe, saisissez Cisco123 :



19. Lorsque la liste des serveurs CUCM sur lesquels figure le fichier CTL apparaît, cliquez sur Done (terminé) :



20. Redémarrez les services TFTP et CallManager sur tous les nœuds dans la grappe qui exécutent ces services.
21. Redémarrez tous les téléphones IP afin qu'ils puissent obtenir la nouvelle version du fichier CTL à partir du serveur TFTP de CUCM.
22. Afin de vérifier le contenu du fichier CTL, saisissez la commande show ctl dans l'interface de ligne de commande. Dans le fichier CTL, vous pouvez voir les certificats des deux eTokens USB (l'un d'eux est utilisé pour signer le fichier CTL). Voici un exemple de sortie :

```
<#root>

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. Pour ce qui concerne les téléphones IP, vous pouvez vérifier qu'après le redémarrage des

téléphones IP, la version du fichier CTL mis à jour a été téléchargée (la somme de contrôle MD5 correspond lorsqu'elle est comparée au résultat de CUCM) :

Cette modification est possible, car vous avez préalablement exporté et chargé les certificats eToken vers la banque de certificats de confiance CUCM, et les téléphones IP peuvent vérifier ce certificat inconnu qui a été utilisé afin de signer le fichier CTL par rapport au Trust Verification Service (service de vérification de confiance) qui s'exécute sur le CUCM.

Cet extrait de journalisation montre comment le téléphone IP contacte les TVS de CUCM avec une requête pour vérifier le certificat eToken inconnu, qui est chargé en tant que Phone-SAST-trust et est approuvé :

<#root>

//

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
```

Régénération de certificat pour la solution CTL sans jetons

Cette section décrit comment régénérer un certificat de sécurité de grappe CUCM lorsque la solution de CTL avec des jetons est utilisée.

Dans le processus de maintenance CUCM, il arrive que le certificat CallManager du nœud de l'éditeur CUCM change.

Les scénarios dans lesquels cela peut se produire sont les suivants : changement de nom d'hôte, changement de domaine ou simple régénération de certificat (en raison de la date d'expiration du certificat).

Une fois le fichier CTL mis à jour, il est signé avec un certificat différent de celui qui existe dans le

fichier CTL qui est installé sur les téléphones IP.

Normalement, ce nouveau fichier CTL n'est pas accepté ; cependant, une fois que le téléphone IP a trouvé le certificat inconnu utilisé pour signer le fichier CTL, il contacte le service TVS sur le CUCM.

Remarque : la liste des serveurs TVS se trouve dans le fichier de configuration du téléphone IP et est mappée dans les serveurs CUCM à partir du pool de périphériques téléphoniques IP > groupe CallManager.

Une fois la vérification effectuée par rapport au serveur TVS, le téléphone IP met à jour son fichier CTL avec la nouvelle version. Ces événements se produisent dans le scénario suivant :

1. Le fichier CTL existe sur CUCM et sur le téléphone IP. Le certificat CCM+TFT (serveur) pour le nœud de l'éditeur CUCM est utilisé pour signer le fichier CTL :

<#root>

```
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

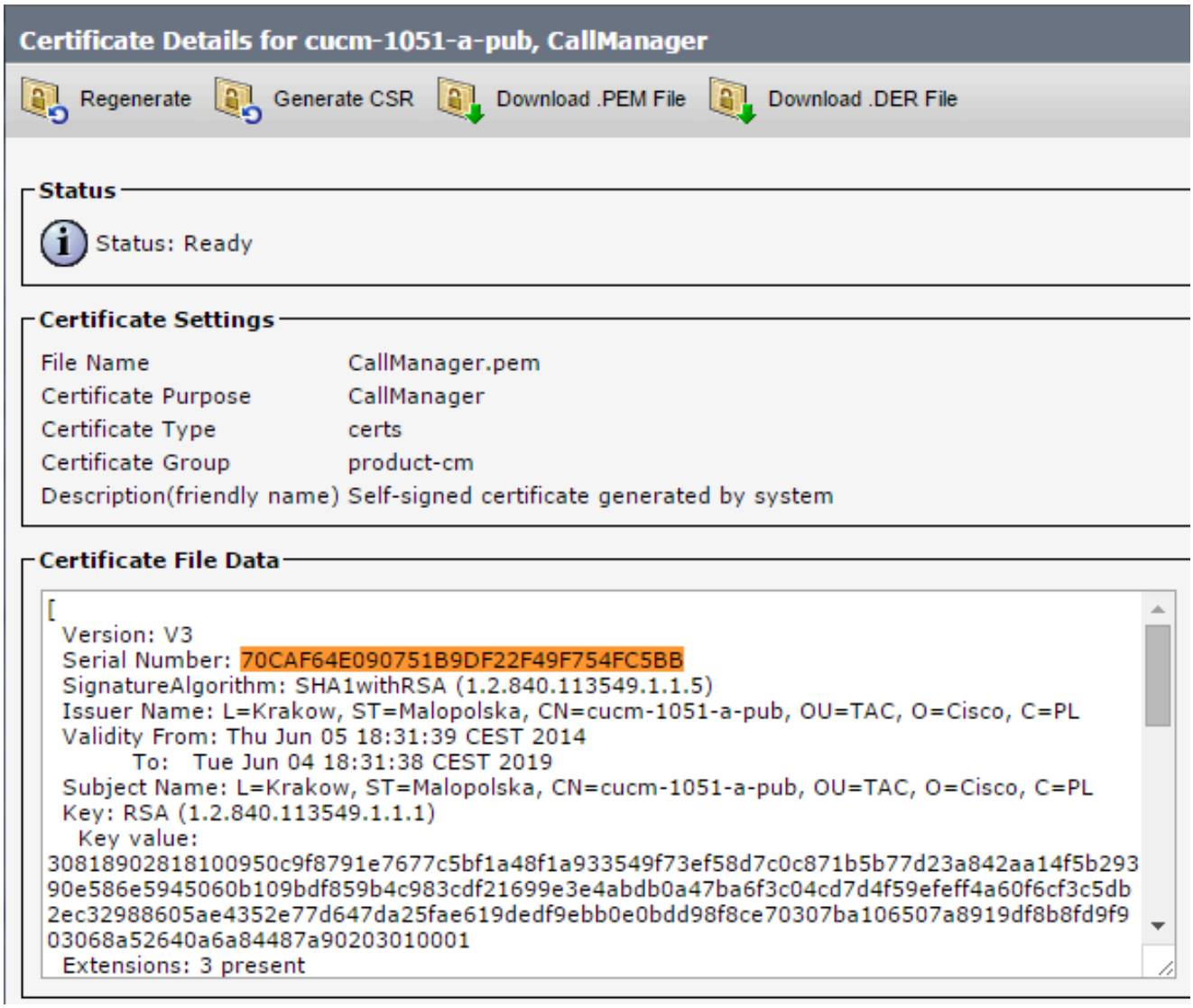
```
5       ISSUERNAME      62      CN=cucm-1051-a-pub;OU=TAC;O=Cisco;L=Krakow;
                                ST=Malopolska;C=PL
6       SERIALNUMBER    16
```

70:CA:F6:4E:09:07:51:B9:DF:22:F4:9F:75:4F:C5:BB

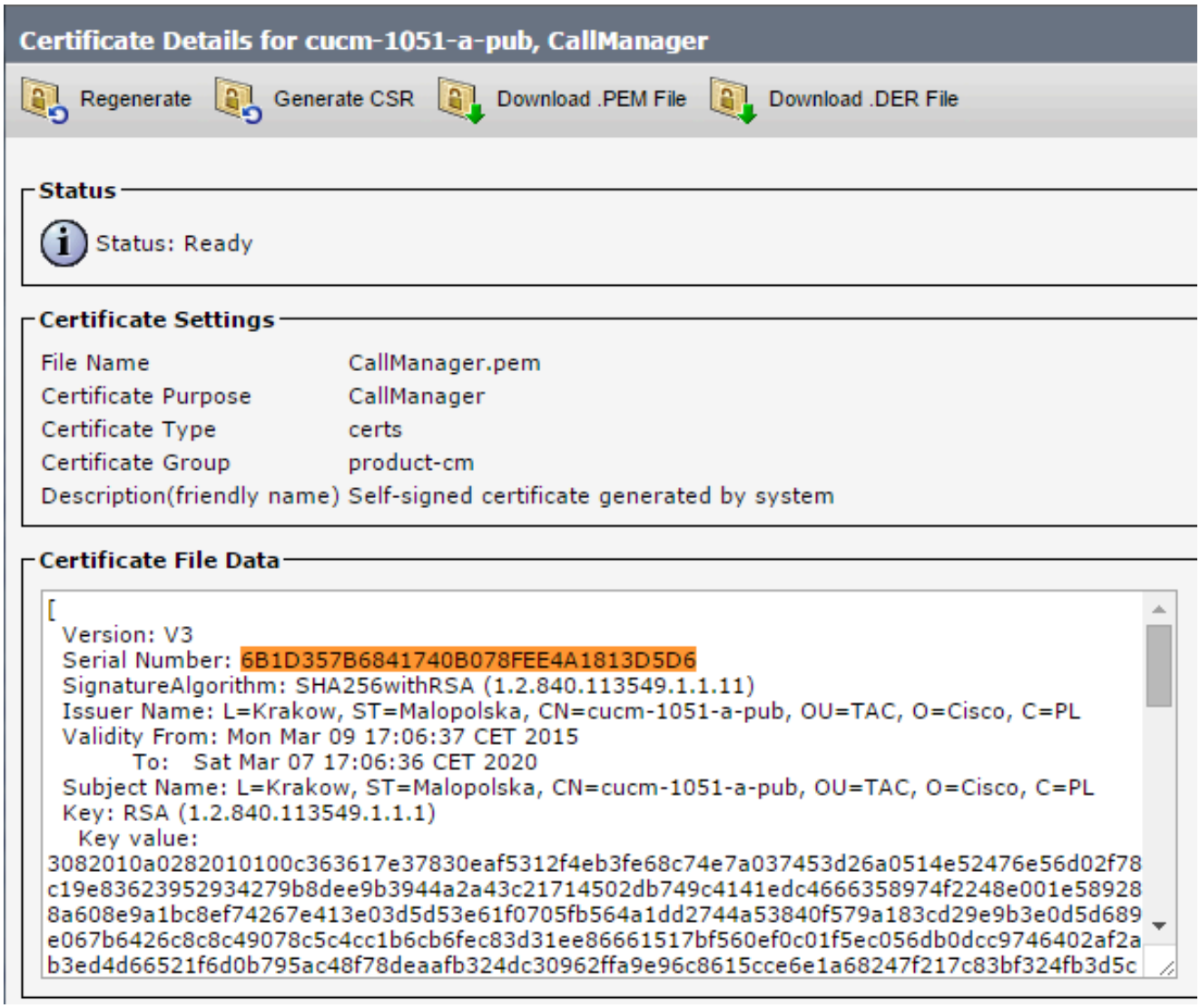
```
7       PUBLICKEY       140
8       SIGNATURE       128
9       CERTIFICATE     694     E9 D4 33 64 5B C8 8C ED 51 4D 8F E5 EA 5B 6D
                                21 A5 A3 8C 9C (SHA1 Hash HEX)
10      IPADDRESS       4
```

[...]

```
The CTL file was verified successfully.
```



2. Le fichier CallManager.pem (certificat CCM+TFTP) est régénéré et vous pouvez voir que le numéro de série du certificat change :



3. La commande utils ctl update CTLFile est entrée dans la CLI afin de mettre à jour le fichier CTL :

<#root>

```
admin:

utils ctl update CTLFile


This operation updates 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. Le service TVS met à jour sa mise en mémoire cache de certificats avec les nouveaux détails du fichier CTL :

<#root>

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 ::

6B1D357B6841740B078FEE4A1813D5D6

CN=

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 ::

6B1D357B6841740B078FEE4A1813D5D6

CN=

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. Lorsque vous affichez le contenu du fichier CTL, vous pouvez voir que le fichier est signé avec le nouveau certificat de serveur CallManager pour le nœud d'éditeur :

<#root>

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
```

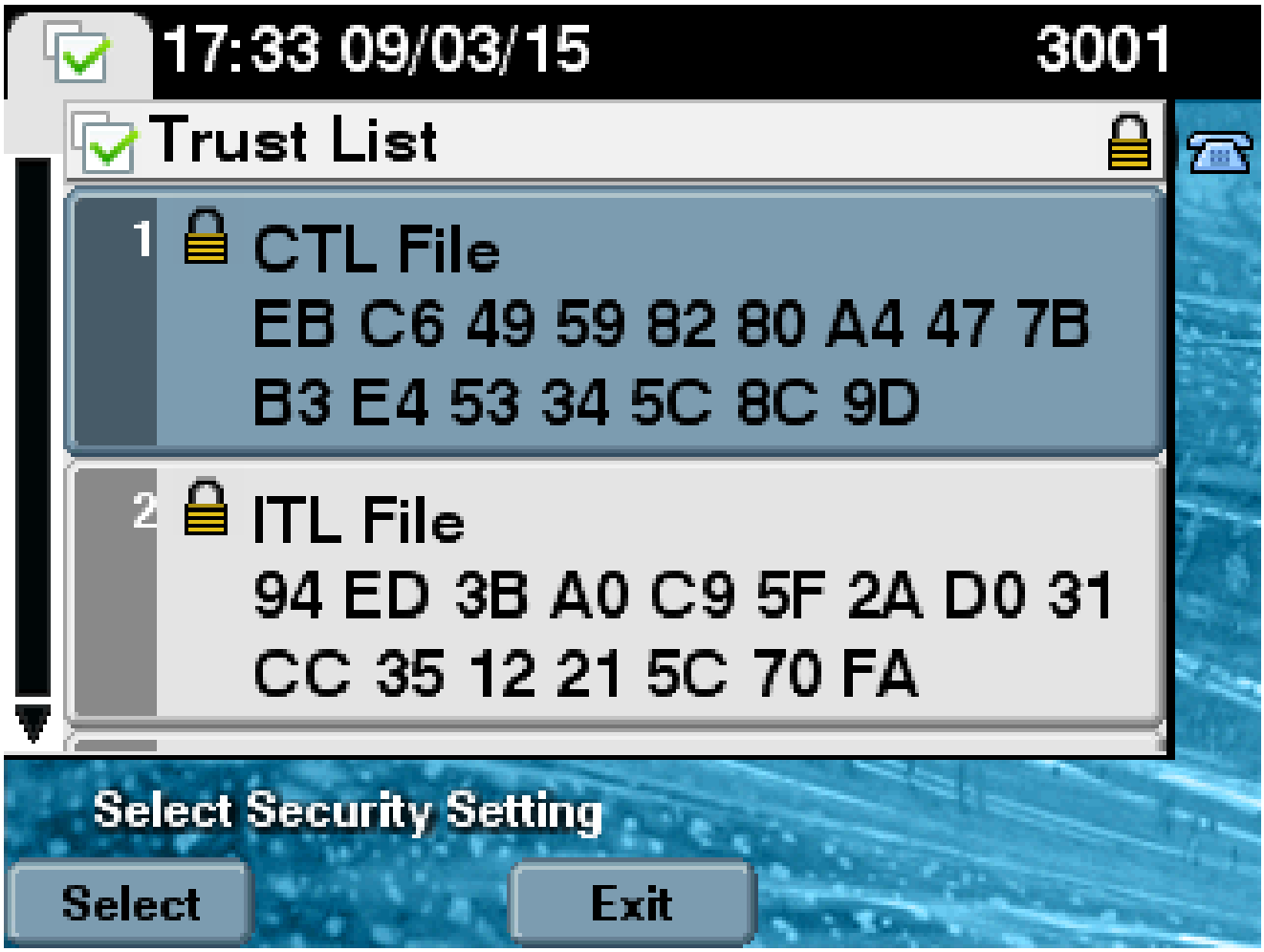
```
8       SIGNATURE       256
9       CERTIFICATE     955     5C AF 7D 23 FE 82 DB 87 2B 6F 4D B7 F0 9D D5
                                86 EE E0 8B FC (SHA1 Hash HEX)
10      IPADDRESS       4


[...]


The CTL file was verified successfully.
```

6. À partir de la page Unified Serviceability, les services TFTP et Cisco CallManager sont redémarrés sur tous les nœuds dans la grappe qui exécutent ces services.
7. Les téléphones IP sont redémarrés et ils communiquent avec le serveur des TVS afin de vérifier le certificat inconnu qui est maintenant utilisé pour signer la nouvelle version du fichier CTL :

```
<#root>

//

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. Enfin, sur les téléphones IP, vous pouvez vérifier que le fichier CTL est mis à jour avec la nouvelle version et que la somme de contrôle MD5 du nouveau fichier CTL correspond à celle de CUCM :



À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.