

# CUCM-Gemischt mit Tokenless CTL

## Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Hintergrundinformationen](#)

[Vom ungesicherten Modus zum gemischten Modus \(Tokenless CTL\)](#)

[Von Hardware-eTokens zu Tokenless-Lösungen](#)

[Von Tokenless-Lösung zu Hardware-eToken](#)

[Zertifikatsgenerierung für Tokenless CTL-Lösung](#)

## Einführung

In diesem Dokument wird der Unterschied zwischen der Cisco Unified Communications Manager-Sicherheit (CUCM) mit und ohne Verwendung von Hardware-USB-eToken beschrieben. In diesem Dokument werden auch die grundlegenden Implementierungsszenarien beschrieben, die Tokenless Certificate Trust List (CTL) umfassen, sowie der Prozess, der verwendet wird, um sicherzustellen, dass das System nach den Änderungen ordnungsgemäß funktioniert.

## Voraussetzungen

### Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse der CUCM-Version 10.0(1) oder höher verfügen. Stellen Sie außerdem sicher, dass:

- Der Lizenzserver für CUCM Version 11.5.1SU3 und höher muss Cisco Prime License Manager (PLM) 11.5.1SU2 oder höher sein. Der Grund hierfür ist, dass die CUCM-Version 11.5.1SU3 die Verschlüsselungslizenz für den gemischten Modus erfordert und PLM die Verschlüsselungslizenz erst ab 11.5.1SU2 unterstützt. Weitere Informationen finden Sie in den [Versionshinweisen für Cisco Prime License Manager, Version 11.5\(1\)SU2](#).
- Sie haben Administratorzugriff auf die Befehlszeilenschnittstelle (CLI) des CUCM Publisher-Knotens.
- Sie haben Zugriff auf die Hardware-USB eToken und dass das CTL-Client Plugin auf Ihrem PC für Szenarien installiert ist, die Sie zur Migration zurück auf die Verwendung von Hardware eTokens benötigen. Zur weiteren Klarstellung: Diese Anforderung gilt nur, wenn Sie an einem beliebigen Punkt ein Szenario haben, in dem USB eToken benötigt werden. Die Wahrscheinlichkeit ist sehr gering, dass USB eToken für die meisten Menschen benötigt werden.

- Alle CUCM-Knoten im Cluster sind vollständig verbunden. Dies ist sehr wichtig, da die CTL-Datei über das SSH File Transfer Protocol (SFTP) auf alle Knoten im Cluster kopiert wird.
- Die Datenbankreplikation (DB) im Cluster funktioniert ordnungsgemäß und die Server replizieren die Daten in Echtzeit.
- Die Geräte in Ihrer Bereitstellung unterstützen Security by Default (TVS). Sie können die *Unified CM-Telefonfunktionsliste* auf der Cisco Unified Reporting-Webseite (<https://<CUCM IP oder FQDN>/cucreports/>) verwenden, um die Geräte zu bestimmen, die standardmäßig Sicherheit unterstützen.

**Hinweis:** Cisco Jabber und viele Cisco TelePresence- oder Cisco IP-Telefone der Serien 7940/7960 unterstützen derzeit standardmäßig keine Sicherheit. Wenn Sie Tokenless CTL mit Geräten bereitstellen, die Security by Default nicht unterstützen, verhindert jedes Update Ihres Systems, das das CallManager-Zertifikat des Herausgebers ändert, die normale Funktionalität dieser Geräte, bis die CTL manuell gelöscht wird. Geräte, die standardmäßig Security by Default unterstützen, z. B. Telefone der Serien 7945 und 7965 oder höher, können CTL-Dateien installieren, wenn das CallManager-Zertifikat auf dem Herausgeber aktualisiert wird, da sie den Trust Verification Service (TVS) verwenden können.

## Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf den folgenden Software- und Hardwareversionen:

- CUCM-Version 10.5.1.1000-7 (Cluster von zwei Knoten)
- Cisco IP-Telefone der Serie 7975, registriert über Skinny Client Control Protocol (SCCP) mit Firmware-Version SCCP75.9-3-1SR4-1S
- Zwei Cisco Security Token, die verwendet werden, um den Cluster mithilfe der CTL-Client-Software auf den gemischten Modus zu setzen

## Hintergrundinformationen

Tokenless CTL ist eine neue Funktion in CUCM-Versionen 10.0(1) und höher, die die Verschlüsselung von Anrufsignalisierung und Medien für IP-Telefone ermöglicht, ohne dass Hardware-USB-eToken und das CTL-Client-Plugin verwendet werden müssen, was in früheren CUCM-Versionen erforderlich war.

Wenn der Cluster mithilfe des CLI-Befehls in den gemischten Modus versetzt wird, wird die CTL-Datei mit dem CCM+TFTP (Server)-Zertifikat des Publisher-Knotens signiert, und in der CTL-Datei sind keine eToken-Zertifikate vorhanden.

**Hinweis:** Wenn Sie das CallManager-Zertifikat (CCM+TFTP) auf dem Herausgeber neu generieren, ändert es den Signator der Datei. Die Telefone und Geräte, die standardmäßig keine Sicherheit unterstützen, akzeptieren die neue CTL-Datei nur, wenn die CTL-Dateien manuell von jedem Gerät gelöscht werden. Weitere Informationen finden Sie im Abschnitt [Anforderungen](#) dieses Dokuments.

# Vom ungesicherten Modus zum gemischten Modus (Tokenless CTL)

In diesem Abschnitt wird der Prozess beschrieben, der verwendet wird, um die CUCM-Clustersicherheit über die CLI in den gemischten Modus zu verschieben.

Vor diesem Szenario befand sich der CUCM im ungesicherten Modus, d. h., dass auf keinem der Knoten eine CTL-Datei vorhanden war und dass auf den registrierten IP-Telefonen nur eine ITL-Datei (Identity Trust List) installiert war, wie in den folgenden Ausgaben gezeigt:

```
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:
```

**Hinweis:** Wenn eine CTL-Datei auf dem Server gefunden wird, während sich der Cluster nicht im gemischten Modus befindet, bedeutet dies, dass der Cluster einmal im gemischten Modus war und dann wieder in den nicht gemischten Modus verschoben wurde und die CTL-Datei nicht aus dem Cluster gelöscht wurde.

Die Befehlsdatei `delete activelog cm/tftpdata/CTLFile.tlv` löscht die CTL-Datei von Knoten im CUCM-Cluster. Der Befehl muss jedoch auf jedem Knoten eingegeben werden. Zur Klarstellung: Verwenden Sie diesen Befehl nur, wenn Ihre Server über eine CTL-Datei verfügen und sich der Cluster nicht im gemischten Modus befindet.

Eine einfache Möglichkeit zu überprüfen, ob sich ein Cluster im gemischten Modus befindet, ist der Befehl `sql select paramname,paramvalue from processing config, wobei paramname='ClusterSecurityMode'`. Wenn der Paramwert 0 ist, befindet sich der Cluster nicht im gemischten Modus.

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



Gehen Sie wie folgt vor, um die CUCM-Cluster-Sicherheit mithilfe der neuen Tokenless CTL-Funktion in den gemischten Modus zu versetzen:

1. Erhalten Sie Administratorzugriff auf die CUCM Publisher-Knoten-CLI.
2. Geben Sie den Befehl `utils ctl set-cluster mixed-mode` in die CLI ein:

```
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. Navigieren Sie zu **CUCM-Admin-Seite > System > Enterprise Parameters**, und überprüfen Sie, ob der Cluster auf den gemischten Modus gesetzt wurde (ein Wert von 1 steht für den gemischten Modus):

Security Parameters	
<u>Cluster Security Mode</u> *	1
<u>LBM Security Mode</u> *	Insecure ▼
<u>CAPF Phone Port</u> *	3804
<u>CAPF Operation Expires in (days)</u> *	10
<u>Enable Caching</u> *	True ▼

4. Starten Sie die TFTP- und Cisco CallManager-Dienste auf allen Knoten im Cluster neu, die diese Dienste ausführen.
5. Starten Sie alle IP-Telefone neu, damit sie die CTL-Datei vom CUCM-TFTP-Dienst beziehen können.

6. Um den Inhalt der CTL-Datei zu überprüfen, geben Sie den Befehl **show ctl** in die CLI ein. In der CTL-Datei sehen Sie, dass das CCM+TFTP (Server)-Zertifikat für den CUCM Publisher-Knoten zum Signieren der CTL-Datei verwendet wird (diese Datei ist auf allen Servern im Cluster identisch). Hier eine Beispielausgabe:

```
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. Auf der Seite des IP-Telefons können Sie überprüfen, ob nach dem Neustart des Diensts die CTL-Datei heruntergeladen wird, die jetzt auf dem TFTP-Server vorhanden ist (die MD5-Prüfsumme stimmt mit der Ausgabe vom CUCM überein):

**Hinweis:** Wenn Sie die Prüfsumme auf dem Telefon überprüfen, sehen Sie **MD5** oder **SHA1**, abhängig vom Telefontyp.



## Von Hardware-eTokens zu Tokenless-Lösungen

In diesem Abschnitt wird beschrieben, wie Sie die CUCM-Cluster-Sicherheit von Hardware-eToken zur Verwendung der neuen Tokenless-Lösung migrieren.

In einigen Fällen ist der gemischte Modus bereits auf dem CUCM mit dem CTL-Client konfiguriert, und die IP-Telefone verwenden CTL-Dateien, die die Zertifikate der Hardware-USB-eToken enthalten. In diesem Szenario wird die CTL-Datei von einem Zertifikat eines USB-eTokens signiert und auf den IP-Telefonen installiert. Hier ein Beispiel:

```
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 ISSUERNAM 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.
```



Gehen Sie wie folgt vor, um die CUCM-Cluster-Sicherheit auf die Verwendung von Tokenless CTLs zu verschieben:

1. Erhalten Sie Administratorzugriff auf die CUCM Publisher-Knoten-CLI.
2. Geben Sie den Befehl `utils ctl update CTLFile` CLI ein:

```
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. Starten Sie die TFTP- und CallManager-Dienste auf allen Knoten im Cluster neu, die diese Dienste ausführen.
4. Starten Sie alle IP-Telefone neu, damit sie die CTL-Datei vom CUCM-TFTP-Dienst beziehen können.
5. Geben Sie den Befehl `show ctl` in die CLI ein, um den Inhalt der CTL-Datei zu überprüfen. In der CTL-Datei können Sie sehen, dass das CCM+TFTP (Server)-Zertifikat des CUCM Publisher-Knotens zum Signieren der CTL-Datei anstelle des Zertifikats der Hardware-USB-eToken verwendet wird. Ein weiterer wichtiger Unterschied in diesem Fall ist, dass die Zertifikate aller USB eTokens der Hardware aus der CTL-Datei entfernt werden. Hier eine Beispielausgabe:

```
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.

**Hinweis:** Wenn in der obigen Ausgabe das CCM+TFTP-Zertifikat (Server) des CUCM Publisher nicht signiert ist, kehren Sie zum hardwarebasierten Cluster-Sicherheitsmodus zurück, und wiederholen Sie die Änderungen für die automatische Lösung erneut.

6. Auf der Seite des IP-Telefons können Sie überprüfen, ob die IP-Telefone nach dem Neustart die aktualisierte CTL-Dateiversion heruntergeladen haben (die MD5-Prüfsumme stimmt mit der Ausgabe des CUCM überein):





## Von Tokenless-Lösung zu Hardware-eToken

In diesem Abschnitt wird beschrieben, wie die CUCM-Cluster-Sicherheit von der neuen Tokenless-Lösung und zurück zur Verwendung von Hardware-eToken migriert wird.

Wenn die CUCM-Cluster-Sicherheit unter Verwendung der CLI-Befehle auf den gemischten Modus gesetzt und die CTL-Datei mit dem CCM+TFTP (Server)-Zertifikat für den CUCM Publisher-Knoten signiert wird, sind in der CTL-Datei keine Zertifikate von den Hardware-USB-eToken vorhanden. Aus diesem Grund wird beim Ausführen des CTL-Clients zur Aktualisierung der CTL-Datei (Zurücksetzen auf die Verwendung von Hardware-eTokens) folgende Fehlermeldung angezeigt:

```
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.
```

Dies ist besonders in Szenarien von Bedeutung, die ein Downgrade (wenn die Version zurückgeschaltet wird) des Systems auf eine Version vor 10.x beinhalten, die die `utils ctl`-Befehle nicht enthält. Die vorherige CTL-Datei wird (ohne inhaltliche Änderungen) während einer Aktualisierung oder eines Upgrades von Linux auf Linux (L2) migriert und enthält, wie bereits erwähnt, keine eToken-Zertifikate. Hier eine Beispielausgabe:

```
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 ISSUERNAM 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 ISSUERNAM 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:

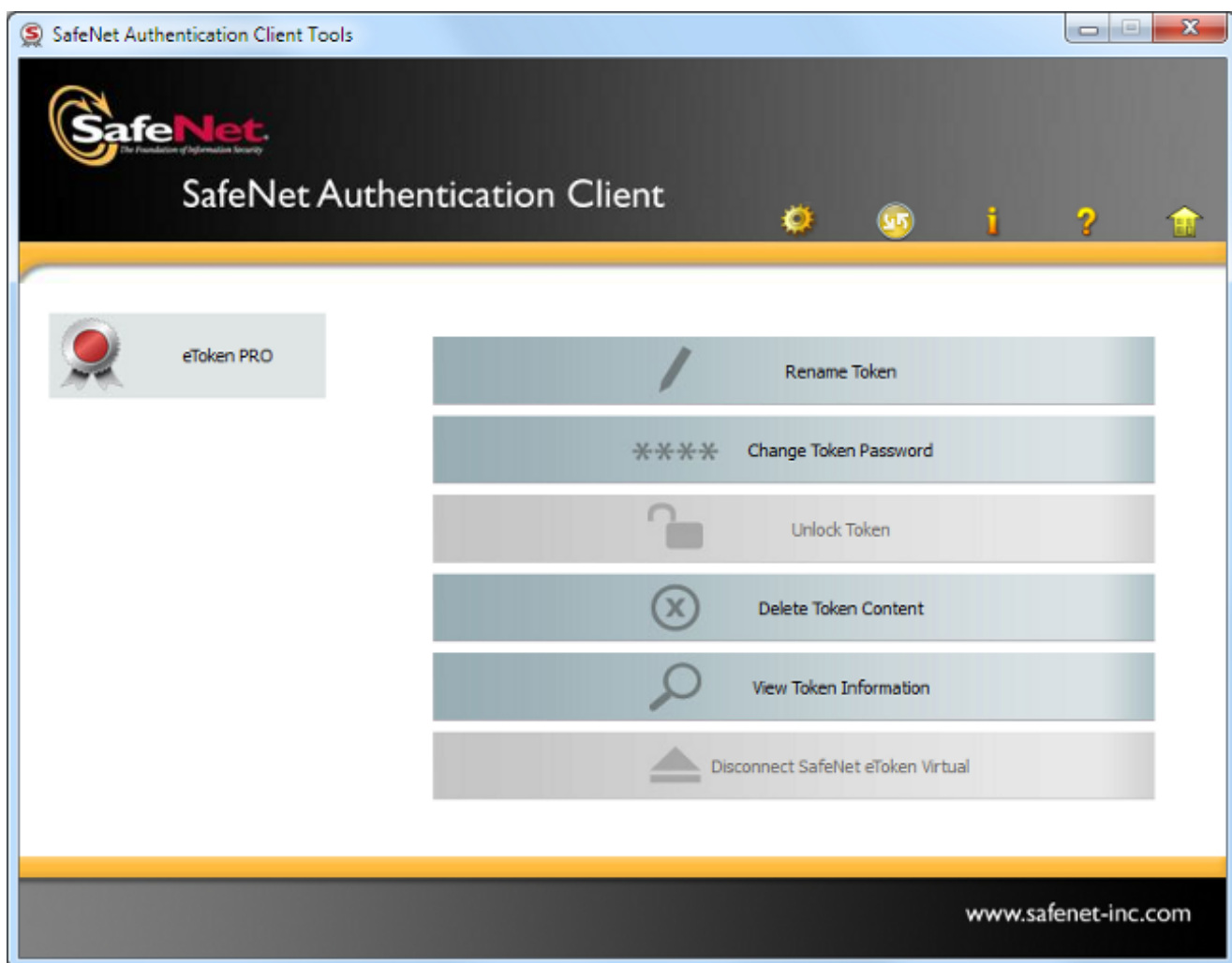
Führen Sie für dieses Szenario die folgenden Schritte aus, um die CTL-Dateien sicher zu aktualisieren, ohne dass das Verfahren für verlorene eToken verwendet werden muss. Diese Datei wird dann manuell aus allen IP-Telefonen gelöscht:

1. Erhalten Sie Administratorzugriff auf die CUCM Publisher-Knoten-CLI.
2. Geben Sie den Befehl **file delete tftp CTLFile.tlv** in die CLI des Publisher-Knotens ein, um die CTL-Datei zu löschen:

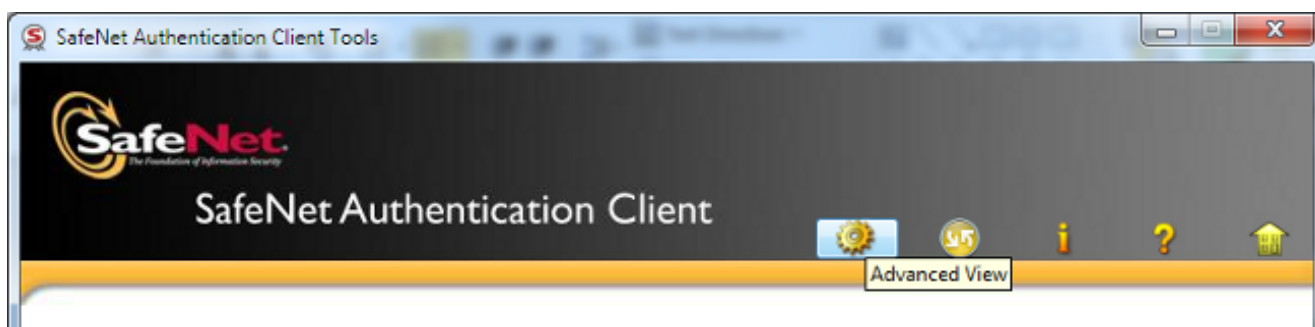
```
admin:file delete tftp CTLFile.tlv
Delete the File CTLFile.tlv?
Enter "y" followed by return to continue: y
files: found = 1, deleted = 1
```

3. Öffnen Sie **SafeNet Authentication Client** auf dem Microsoft Windows-Computer, auf dem der

CTL-Client installiert ist (er wird automatisch mit dem CTL-Client installiert):

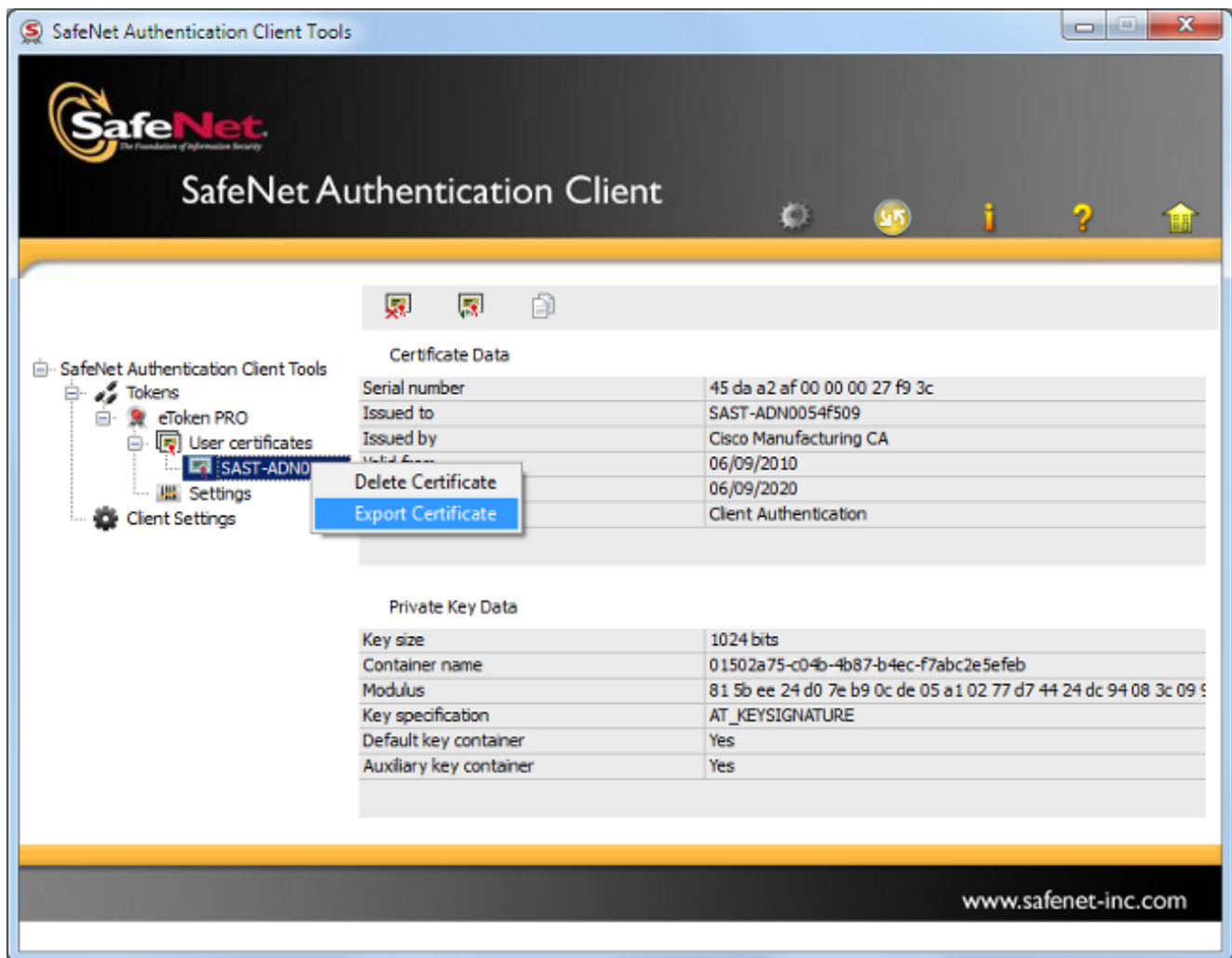


4. Navigieren Sie im SafeNet Authentication Client zur *erweiterten Ansicht*:



5. Stecken Sie das erste Hardware-USB-eToken ein.

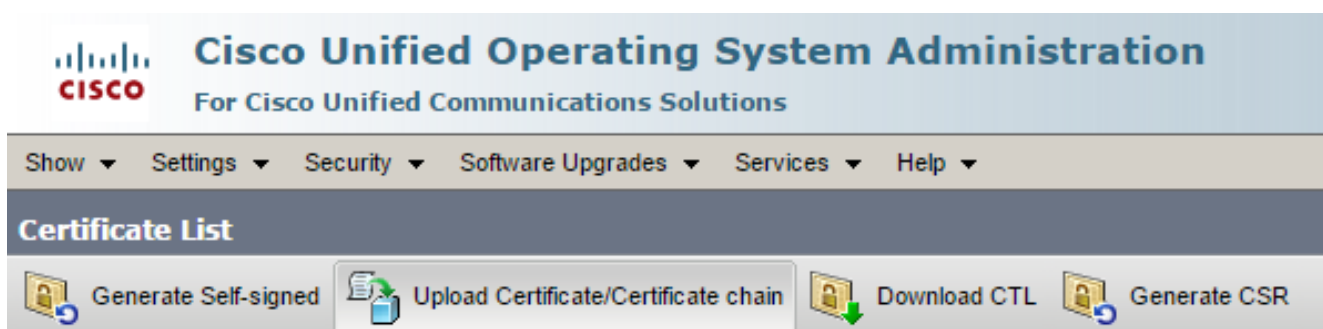
6. Wählen Sie das Zertifikat im Ordner *Benutzerzertifikate aus* und exportieren Sie es in den Ordner auf dem PC. Wenn Sie zur Eingabe eines Kennworts aufgefordert werden, verwenden Sie das Standardkennwort **Cisco123**:



7. Wiederholen Sie diese Schritte für das zweite Hardware-USB-eToken, sodass beide Zertifikate auf den PC exportiert werden:

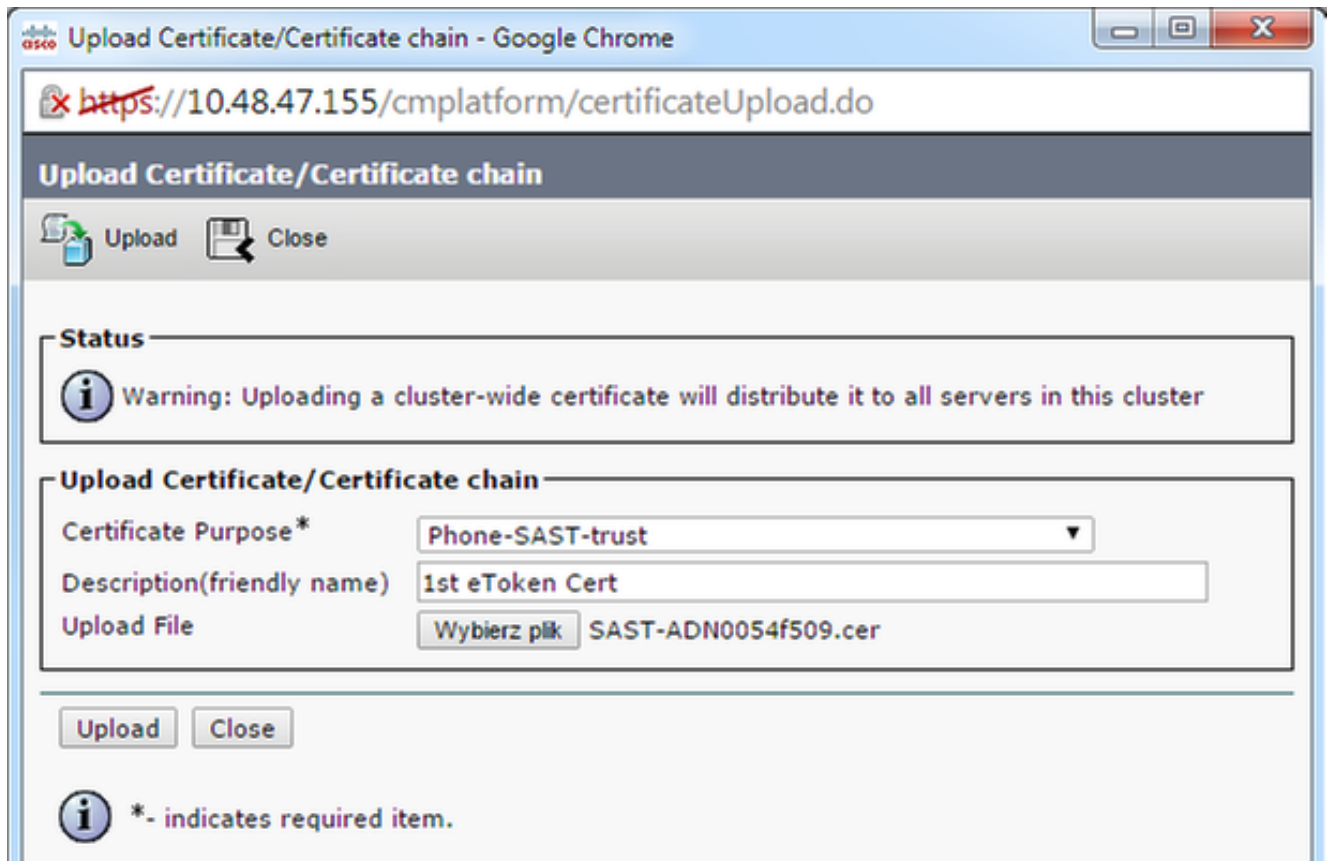
Name	Date modified	Type	Size
SAST-ADN0054f509	06-03-2015 22:32	Security Certificate	1 KB
SAST-ADN008580ef	06-03-2015 22:33	Security Certificate	1 KB

8. Melden Sie sich bei der Cisco Unified Operating System (OS)-Verwaltung an, und navigieren Sie zu **Security > Certificate Management > Upload Certificate**:

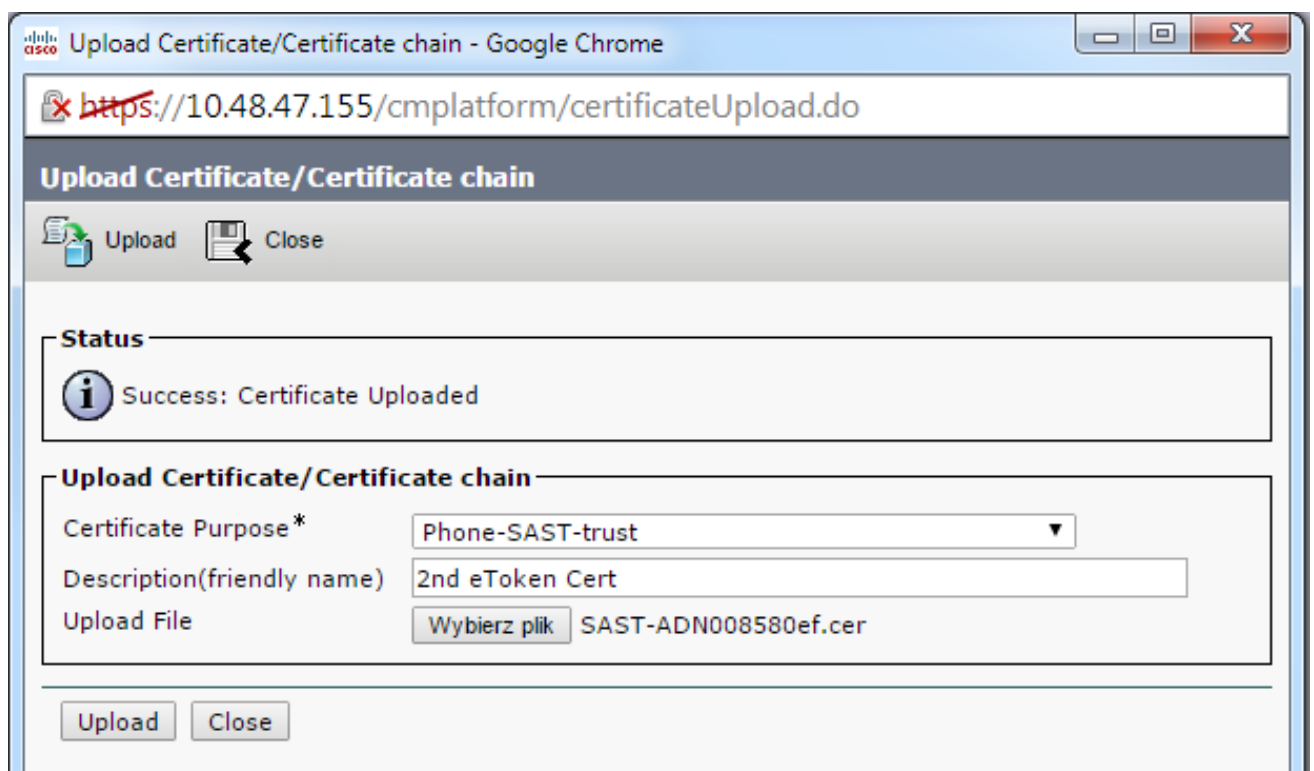


9. Dann wird die Seite Zertifikat hochladen angezeigt. Wählen Sie **Phone-SAST-trust** im Dropdown-Menü "Certificate Purpose" (Zweck des Zertifikats) aus, und wählen Sie das

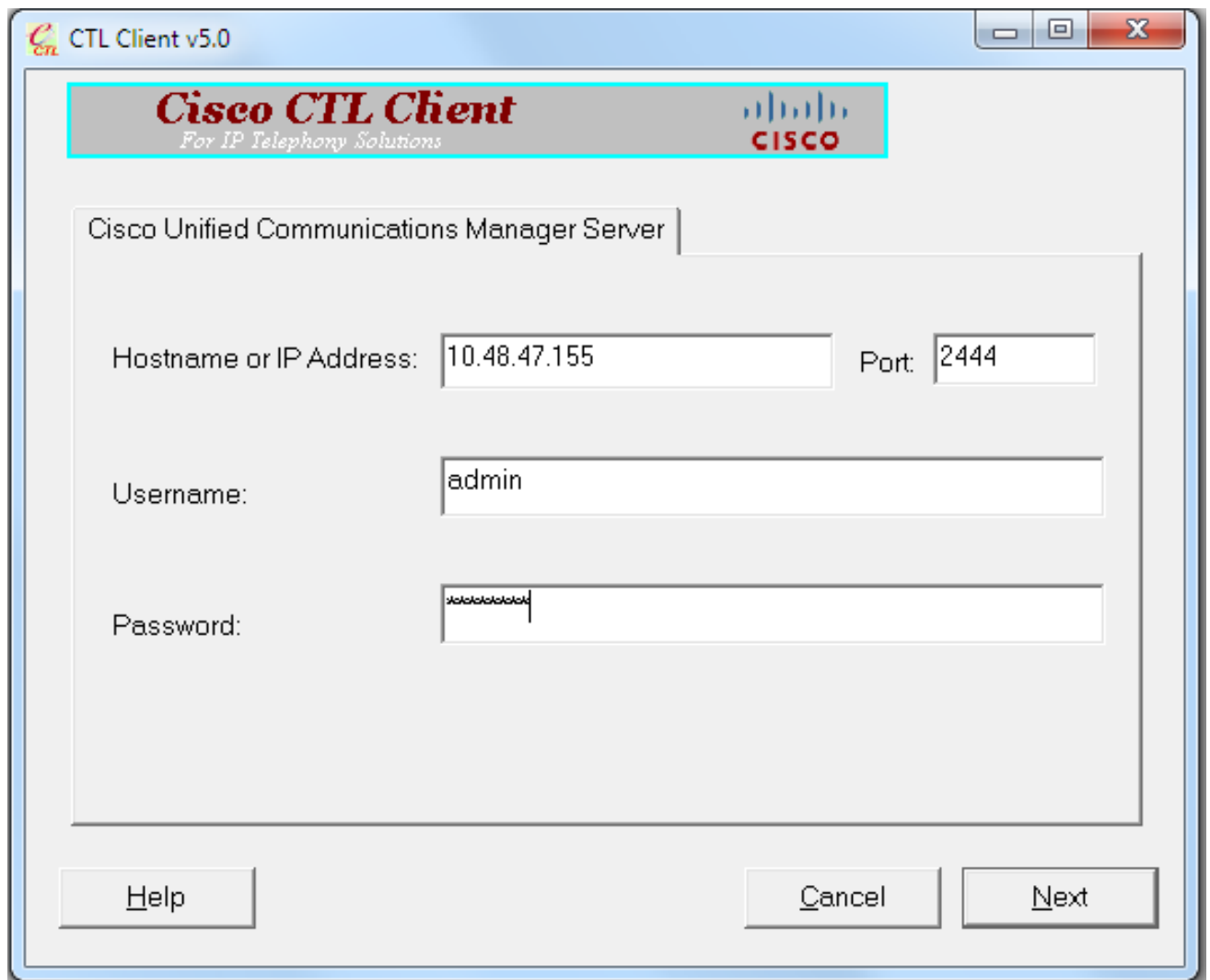
Zertifikat aus, das Sie aus dem ersten eToken exportiert haben:



10. Führen Sie die vorherigen Schritte aus, um das vom zweiten eToken exportierte Zertifikat hochzuladen:



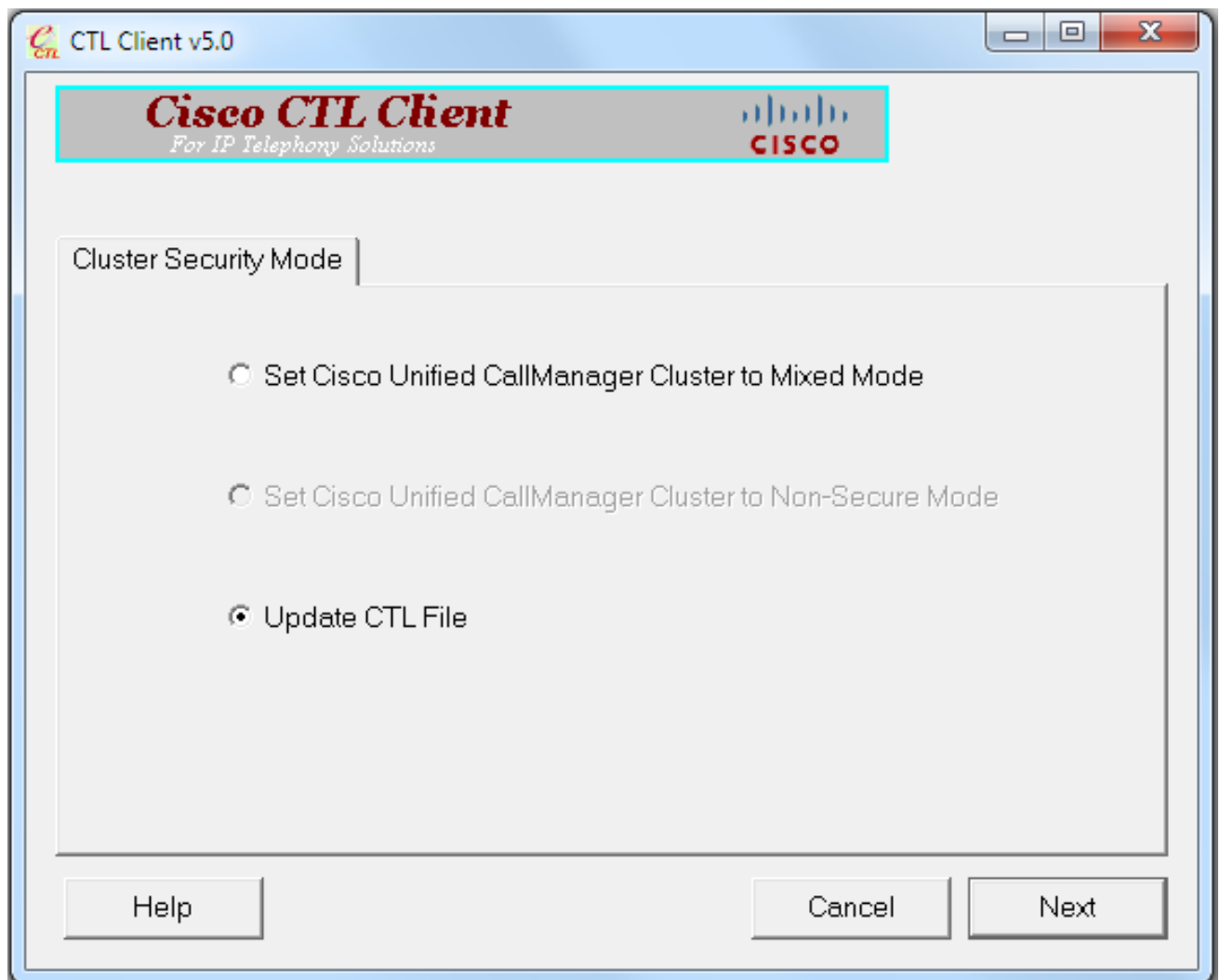
11. Führen Sie den CTL-Client aus, geben Sie die IP-Adresse/den Hostnamen des CUCM Publisher-Knotens ein, und geben Sie die CCM-Administratoranmeldeinformationen ein:



12. Da sich der Cluster bereits im gemischten Modus befindet, jedoch keine CTL-Datei auf dem Publisher-Knoten vorhanden ist, wird diese Warnmeldung angezeigt (klicken Sie auf **OK**, um sie zu ignorieren):

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. Klicken Sie im CTL-Client auf das Optionsfeld **CTL-Datei aktualisieren**, und klicken Sie dann auf **Weiter**:

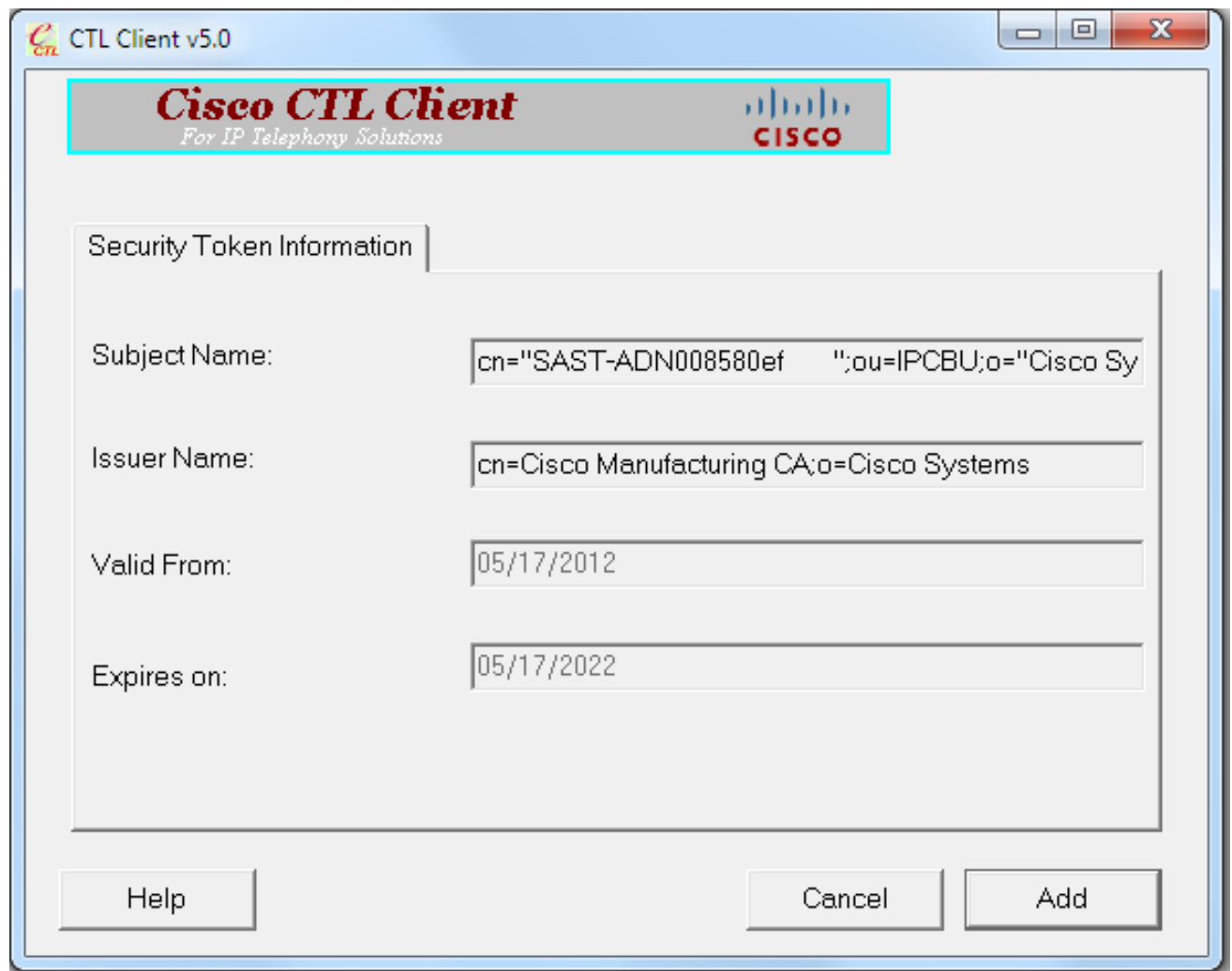


14. Legen Sie das erste Sicherheitstoken ein, und klicken Sie auf **OK**:

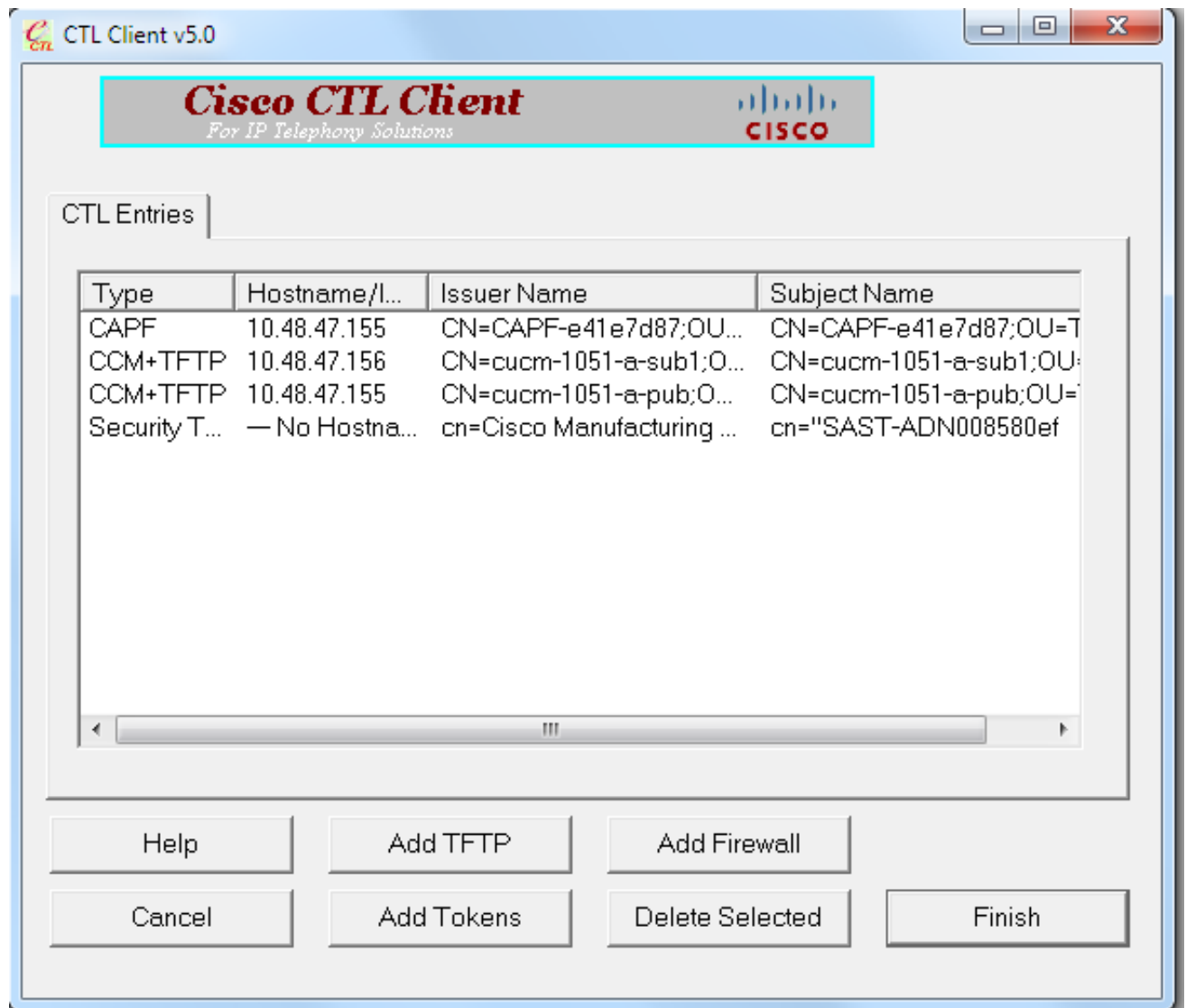


15. Wenn die Details des Sicherheitstokens angezeigt werden, klicken Sie auf **Hinzufügen**:

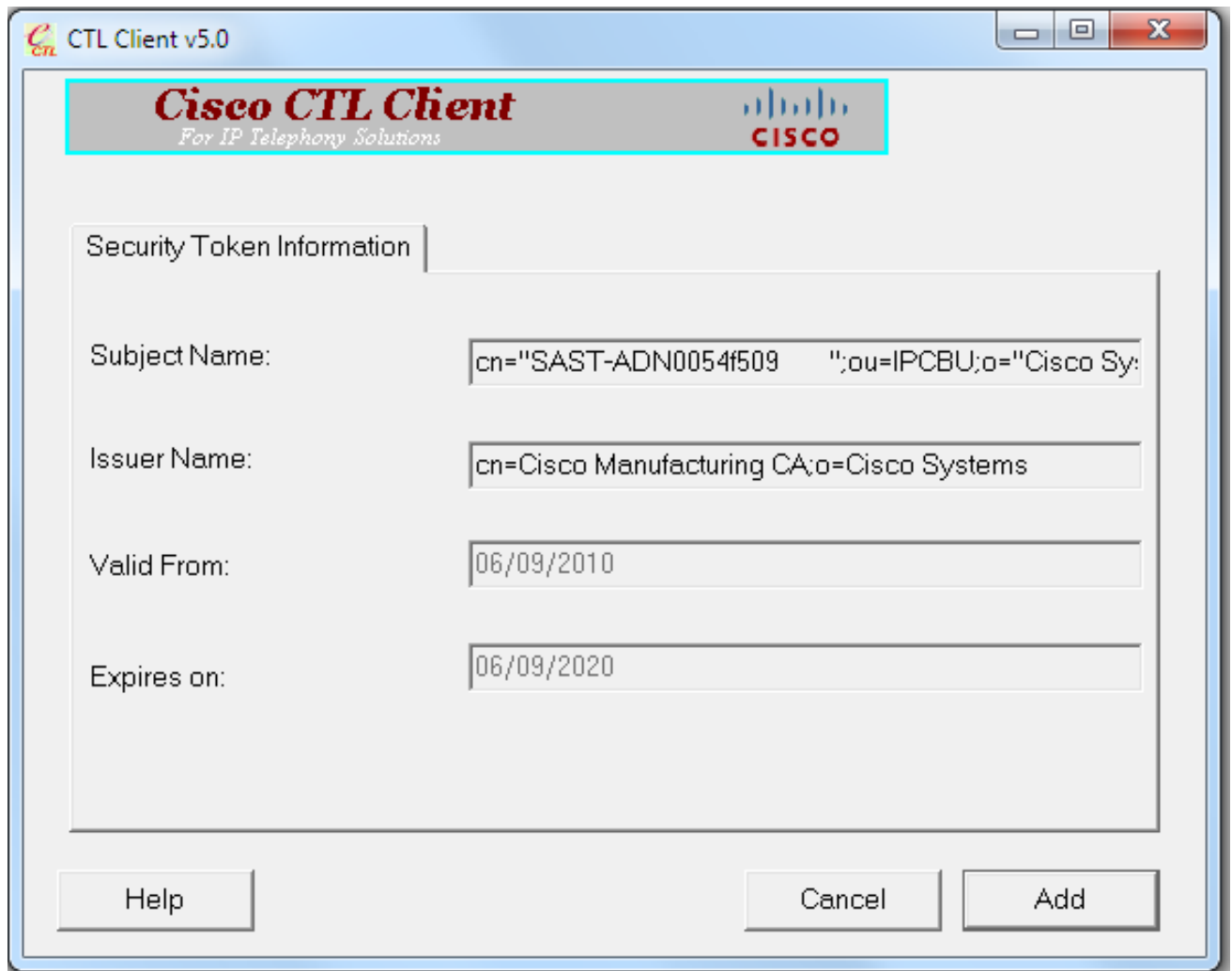




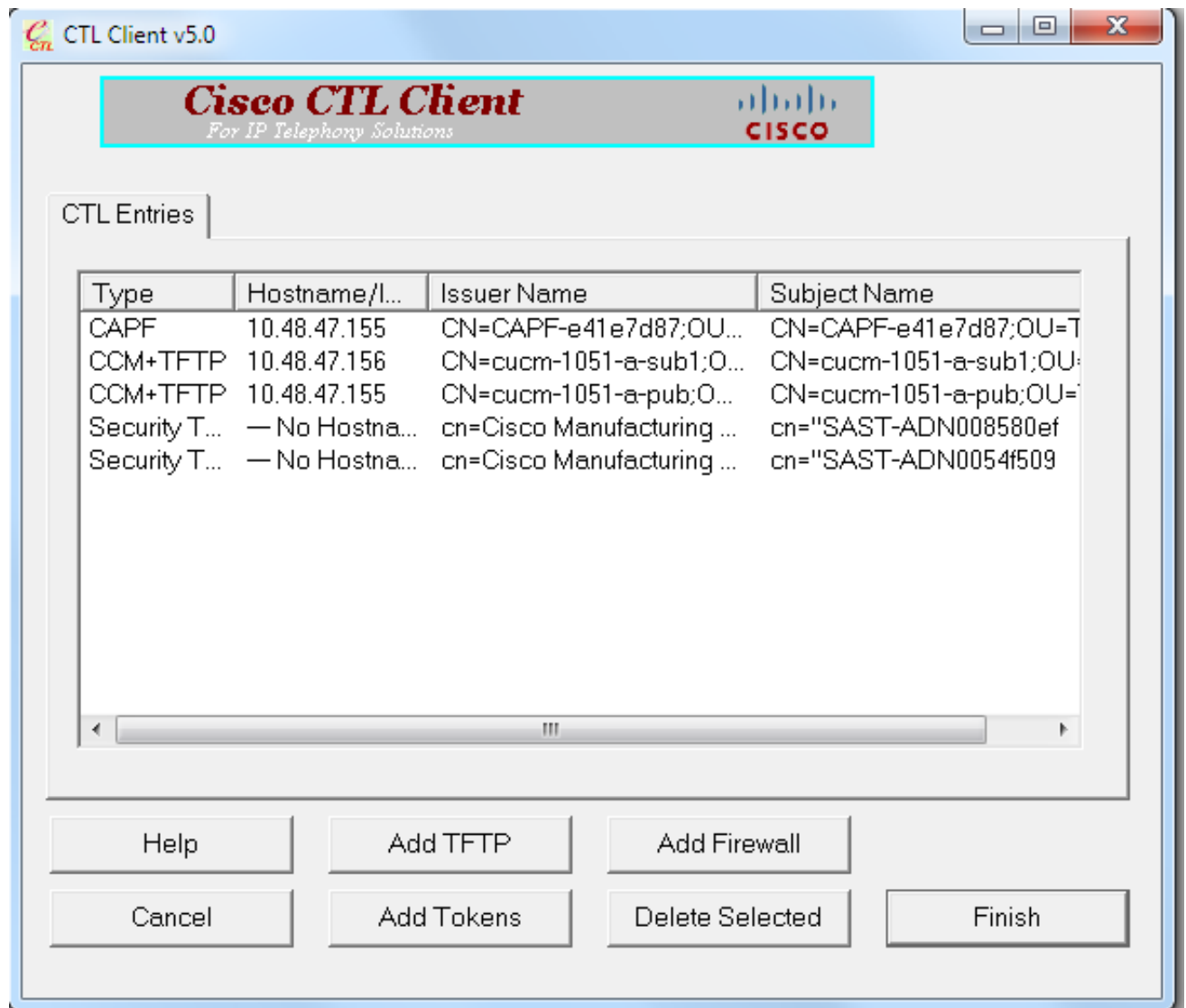
16. Sobald der Inhalt der CTL-Datei angezeigt wird, klicken Sie auf **Token hinzufügen**, um das zweite USB-eToken hinzuzufügen:



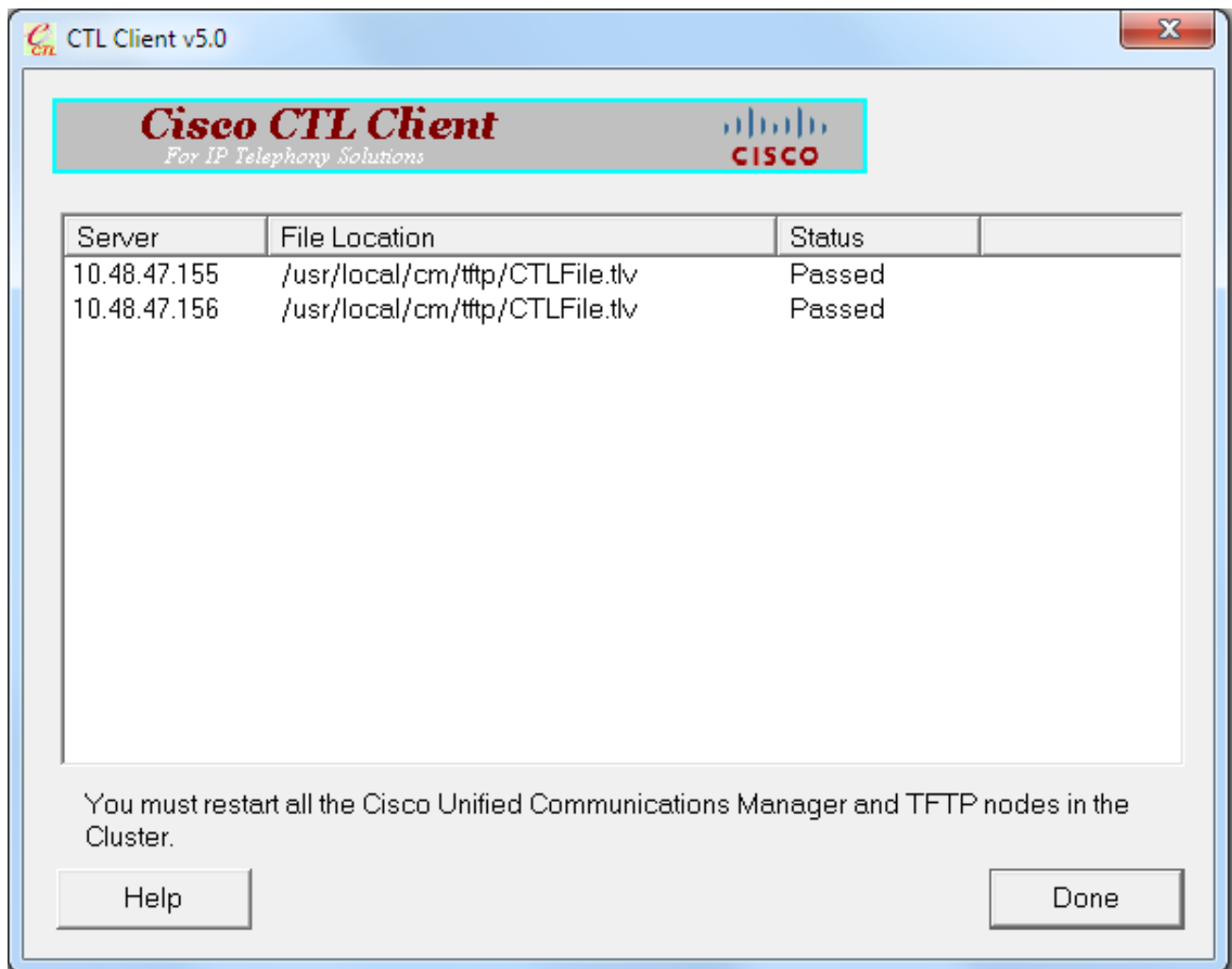
17. Wenn die Details zum Sicherheitstoken angezeigt werden, klicken Sie auf **Hinzufügen**:



18. Wenn der Inhalt der CTL-Datei angezeigt wird, klicken Sie auf **Fertig stellen**. Wenn Sie zur Eingabe eines Kennworts aufgefordert werden, geben Sie **Cisco123 ein**:



19. Wenn die Liste der CUCM-Server angezeigt wird, auf denen die CTL-Datei vorhanden ist, klicken Sie auf **Fertig**:



20. Starten Sie die TFTP- und CallManager-Dienste auf allen Knoten im Cluster neu, die diese Dienste ausführen.
21. Starten Sie alle IP-Telefone neu, damit sie die neue Version der CTL-Datei vom CUCM TFTP-Dienst erhalten.
22. Um den Inhalt der CTL-Datei zu überprüfen, geben Sie den Befehl **show ctl** in die CLI ein. In der CTL-Datei werden die Zertifikate beider USB-eToken angezeigt (eines davon wird zum Signieren der CTL-Datei verwendet). Hier eine Beispielausgabe:

```

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 ISSUENAME 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 ISSUENAME 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. Auf der Seite des IP-Telefons können Sie überprüfen, ob die IP-Telefone nach dem Neustart die aktualisierte CTL-Dateiversion heruntergeladen haben (die MD5-Prüfsumme stimmt mit der Ausgabe des CUCM überein):



Diese Änderung ist möglich, da Sie die eToken-Zertifikate zuvor exportiert und in den CUCM Certificate Trust Store hochgeladen haben. Die IP-Telefone können das unbekannte Zertifikat überprüfen, das zum Signieren der CTL-Datei gegen den Trust Verification Service (TVS) verwendet wurde, der auf dem CUCM ausgeführt wird. Dieser Protokollausschnitt veranschaulicht, wie das IP-Telefon mit dem CUCM TVS eine Anforderung zur Verifizierung des unbekannten eToken-Zertifikats, das als **Phone-SAST-trust** hochgeladen und als vertrauenswürdig hochgeladen wird, kontaktiert:

**//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: 83E908000005545AF31
23:00:22.052 | debug CertificateDBCACHE::getCertificateInformation - Looking up the
certificate cache using Unique MAP ID : 83E908000005545AF31cn=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 : 83E908000005545AF31cn=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
```

## Zertifikatsgenerierung für Tokenless CTL-Lösung

In diesem Abschnitt wird beschrieben, wie Sie ein CUCM-Cluster-Sicherheitszertifikat neu generieren, wenn die Tokenless CTL-Lösung verwendet wird.

Im Zuge der CUCM-Wartung wird manchmal das CallManager-Zertifikat des CUCM Publisher-Knotens geändert. Zu den Szenarien, in denen dies passieren kann, gehören die Änderung des Hostnamens, die Änderung der Domäne oder einfach eine Erneuerung des Zertifikats (aufgrund des Ablaufdatums des Zertifikats).

Nachdem die CTL-Datei aktualisiert wurde, wird sie mit einem anderen Zertifikat signiert als das in der CTL-Datei, die auf den IP-Telefonen installiert ist. Diese neue CTL-Datei wird normalerweise nicht akzeptiert. Nachdem das IP-Telefon jedoch das unbekannte Zertifikat gefunden hat, das zum Signieren der CTL-Datei verwendet wird, kontaktiert es den TVS-Dienst auf dem CUCM.

**Hinweis:** Die Liste der TVS-Server befindet sich in der Konfigurationsdatei für das IP-Telefon und wird den CUCM-Servern aus dem IP-Telefon-**Gerätepool > CallManager Group** zugeordnet.

Nach erfolgreicher Verifizierung des TVS-Servers aktualisiert das IP-Telefon seine CTL-Datei mit der neuen Version. Diese Ereignisse treten in einem solchen Szenario auf:

1. Die CTL-Datei ist auf dem CUCM und dem IP-Telefon vorhanden. Das CCM+TFT (Server)-Zertifikat für den CUCM Publisher-Knoten wird zum Signieren der CTL-Datei verwendet:

```
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.

### Certificate Details for cucm-1051-a-pub, CallManager

Regenerate Generate CSR Download .PEM File Download .DER File

---

**Status**

 Status: Ready

---

**Certificate Settings**

File Name	CallManager.pem
Certificate Purpose	CallManager
Certificate Type	certs
Certificate Group	product-cm
Description(friendly name)	Self-signed certificate generated by system





---

**Certificate File Data**

```
[
Version: V3
Serial Number: 70CAF64E090751B9DF22F49F754FC5BB
SignatureAlgorithm: SHA1withRSA (1.2.840.113549.1.1.5)
Issuer Name: L=Krakow, ST=Malopolska, CN=cucm-1051-a-pub, OU=TAC, O=Cisco, C=PL
Validity From: Thu Jun 05 18:31:39 CEST 2014
To: Tue Jun 04 18:31:38 CEST 2019
Subject Name: L=Krakow, ST=Malopolska, CN=cucm-1051-a-pub, OU=TAC, O=Cisco, C=PL
Key: RSA (1.2.840.113549.1.1.1)
Key value:
30818902818100950c9f8791e7677c5bf1a48f1a933549f73ef58d7c0c871b5b77d23a842aa14f5b293
90e586e5945060b109bdf859b4c983cdf21699e3e4abdb0a47ba6f3c04cd7d4f59efeff4a60f6cf3c5db
2ec32988605ae4352e77d647da25fae619dedf9ebb0e0bdd98f8ce70307ba106507a8919df8b8fd9f9
03068a52640a6a84487a90203010001
Extensions: 3 present
```


- Die Datei **CallManager.pem** (CCM+TFTP-Zertifikat) wird neu generiert, und die Seriennummer des Zertifikats ändert sich:

### Certificate Details for cucm-1051-a-pub, CallManager

 Regenerate
  Generate CSR
  Download .PEM File
  Download .DER File

---

**Status**

 Status: Ready

---

**Certificate Settings**

File Name	CallManager.pem
Certificate Purpose	CallManager
Certificate Type	certs
Certificate Group	product-cm
Description(friendly name)	Self-signed certificate generated by system

---

**Certificate File Data**

```
[
Version: V3
Serial Number: 6B1D357B6841740B078FEE4A1813D5D6
SignatureAlgorithm: SHA256withRSA (1.2.840.113549.1.1.11)
Issuer Name: L=Krakow, ST=Malopolska, CN=cucm-1051-a-pub, OU=TAC, O=Cisco, C=PL
Validity From: Mon Mar 09 17:06:37 CET 2015
To: Sat Mar 07 17:06:36 CET 2020
Subject Name: L=Krakow, ST=Malopolska, CN=cucm-1051-a-pub, OU=TAC, O=Cisco, C=PL
Key: RSA (1.2.840.113549.1.1.1)
Key value:
3082010a0282010100c363617e37830eaf5312f4eb3fe68c74e7a037453d26a0514e52476e56d02f78
c19e83623952934279b8dee9b3944a2a43c21714502db749c4141edc4666358974f2248e001e58928
8a608e9a1bc8ef74267e413e03d5d53e61f0705fb564a1dd2744a53840f579a183cd29e9b3e0d5d689
e067b6426c8c8c49078c5c4cc1b6cb6fec83d31ee86661517bf560ef0c01f5ec056db0dcc9746402af2a
b3ed4d66521f6d0b795ac48f78deaafb324dc30962ffa9e96c8615cce6e1a68247f217c83bf324fb3d5c
]
```

3. Der Befehl `utils ctl update CTLFile` wird in die CLI eingegeben, um die CTL-Datei zu aktualisieren:

```
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. Der TVS-Dienst aktualisiert seinen Zertifikatscache mit den neuen CTL-Dateidetails:

```
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. Wenn Sie den CTL-Dateiinhalte anzeigen, sehen Sie, dass die Datei mit dem neuen CallManager-Serverzertifikat für den Publisher-Knoten signiert wird:

```
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 ISSUENAME 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 ISSUENAME 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. Auf der Seite für Unified Service-Funktionen werden die TFTP- und Cisco CallManager-Services auf allen Knoten im Cluster neu gestartet, auf denen diese Services ausgeführt

werden.

- Die IP-Telefone werden neu gestartet, und sie wenden sich an den TVS-Server, um das unbekannte Zertifikat zu überprüfen, das jetzt zum Signieren der neuen Version der CTL-Datei verwendet wird:

```
// 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
```

- Schließlich können Sie auf den IP-Telefonen überprüfen, ob die CTL-Datei mit der neuen Version aktualisiert wird und ob die MD5-Prüfsumme der neuen CTL-Datei mit der CUCM-Datei übereinstimmt:

