

CUCM混合模式与Tokenless CTL

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[从不安全的模式到混合模式\(Tokenless CTL\)](#)

[从硬件eTokens到Tokenless解决方案](#)

[从Tokenless解决方案到硬件eTokens](#)

[Tokenless CTL解决方案的证书重新生成](#)

简介

本文描述Cisco Unified Communications Manager (CUCM)安全之间的差异有和没有使用硬件USB eTokens。本文也描述介入Tokenless证书信任列表(CTL)和进程使用为了在更改以后保证系统作用适当地基本实施方案。

[先决条件](#)

[要求](#)

Cisco建议您有CUCM版本10.0(1)或以上知识。另外，请保证那：

- 您访问管理访问CUCM发行商节点的命令行界面(CLI)。
- 您访问硬件USB eTokens，并且那CTL客户端插件在您的要求您移植回到使用硬件eTokens的方案PC安装。
- 有所有的全连接在集群的CUCM节点之间。这是非常重要，因为CTL文件复制对所有在集群的节点通过SSH文件传输协议(SFTP)。
- 在集群的数据库(DB)复制适当地运作，并且那服务器在实时复制数据。
- 在您的部署的设备支持安全默认情况下(TV)。能使用从Cisco Unified的 *Unified CM电话功能列表* 报告网页的您([https:// <CUCM IP或FQDN>/cucreports/](https://<CUCM IP或FQDN>/cucreports/))为了确定默认情况下支持安全的设备。**Note:** 默认情况下Cisco Jabber和许多思科网真或Cisco 7940/7960系列IP电话当前不支持安全。如果部署Tokenless CTL用默认情况下不支持安全的设备，对更改的您的系统的所有更新

在发行商的CallManager证书将防止那些设备注册用系统，直到他们的CTL手工删除。

使用的组件

本文档中的信息基于以下软件和硬件版本：

- CUCM版本10.5.1.10000-7 (两节点集群)
- 通过与固件版本SCCP75.9-3-1SR4-1S的小型客户机控制协议(SCCP)注册的Cisco 7975系列IP电话
- 用于为了设置集群到混合模式与使用CTL客户端软件的两个Cisco安全令牌

背景信息

Tokenless CTL是允许呼叫信令和媒体加密IP电话，不用需要使用硬件USB eTokens和插件CTL的客户端，是在上一个CUCM版本的需求在CUCM版本10.0(1)和以上的一新特性。

当集群被放置到混合模式与使用CLI命令时，CTL文件签字与发行商节点的CCM+TFTP (服务器)证书，并且没有eToken证书现在CTL文件。

Note:当您重新生成在发行商的CallManager (CCM+TFTP)时证书，更改文件的签署人。不支持安全默认情况下的电话和设备不会接受新的CTL文件，除非CTL文件从每个设备手工删除。参考列出本文[Requirements部分](#)欲知更多信息的最后要求。

从不安全的模式到混合模式(Tokenless CTL)

此部分描述使用为了搬入CUCM集群安全混合模式通过CLI的进程。

在此方案之前，CUCM在不安全的模式，因此意味着没有CTL文件在任何节点，并且已注册IP电话有安装的仅一个标识托拉斯列表(ITL)文件，如这些输出所显示，：

```
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:
```

为了搬入CUCM集群安全混合模式与使用新的Tokenless CTL功能，请完成这些步骤：

1. 获取对CUCM发行商节点CLI的管理访问。
2. 输入使用情况ctl设置团星mixed-mode命令到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. 导航对CUCM管理员页面>System >企业参数并且验证集群是否设置对混合模式(值1指示混合模式) :

4. 重新启动TFTP和Cisco CallManager服务在管理这些服务的所有节点在集群。

5. 重新启动所有IP电话，以便他们能从CUCM TFTP服务得到CTL文件。

6. 为了验证CTL文件的内容，请输入显示ctl命令到CLI。在CTL文件中您能看到CUCM发行商节点的CCM+TFTP (服务器)证书用于为了签署CTL文件(此文件是相同的在集群的所有服务器)。以下为示例输出：

```
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 ISSUENAME 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 ISSUENAME 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. 在IP电话侧，您能验证那，在服务被重新启动后，它下载CTL文件，当前是存在TFTP server (MD5校验和配比，当与从CUCM的输出比较)：

Note: 当您验证在电话时的校验和，您看到MD5或SHA1，从属在电话类型。

从硬件eTokens到Tokenless解决方案

此部分描述如何移植从硬件eTokens的CUCM集群安全到使用新的Tokenless解决方案。

在某些状况下，混合模式在与使用的CUCM CTL客户端和包含从硬件USB eTokens的证书的IP电话使用CTL文件已经配置。使用此方案，CTL文件由从其中的一一证书在IP电话签字USB eTokens和安装。此处是示例：

```
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 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.

完成这些步骤为了移动CUCM集群安全向使用Tokenless CTLs：

1. 获取对CUCM发行商节点CLI的管理访问。
2. 输入使用情况ctl更新CLI命令的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. 重新启动TFTP和CallManager服务在管理这些服务的所有节点在集群。
4. 重新启动所有IP电话，以便他们能从CUCM TFTP服务得到CTL文件。
5. 输入**显示ctl**命令到CLI为了验证CTL文件的内容。在CTL文件中，您能看到CUCM发行商节点的CCM+TFTP (服务器)证书用于为了签署CTL文件而不是从硬件USB eTokens的证书。一更加重要的差异在这种情况下是从所有的证书硬件USB eTokens从CTL文件删除。以下为示例输出：

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

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

[...]
```

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

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

[...]
```

The CTL file was verified successfully.

6. 在IP电话侧，您能验证那，在IP电话重新启动后，他们下载更新CTL文件版本(MD5校验和配比，当与从CUCM的输出比较)：

从Tokenless解决方案到硬件eTokens

此部分描述如何移植CUCM集群安全远离新的Tokenless解决方案和回到使用硬件eTokens。

当CUCM集群安全设置对混合模式与使用CLI命令时，并且CTL文件签字与CUCM发行商节点的CCM+TFTP (服务器)证书，没有从硬件USB eTokens的证书现在CTL文件。为此，当您运行CTL客户端为了更新CTL文件(回到使用的移动硬件eTokens)时，此错误消息出现：

```
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 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

[...]

The CTL file was verified successfully.

这是特别重要在包括降级的方案(当版本是交换的上一步)时系统对pre-10.x版本不包括使用情况ctl命令。上一个CTL文件在刷新或Linux过程中被移植(没有在其内容上的变化)到Linux (L2)升级，并且不包含eToken证书，如前所提及。以下为示例输出：

```
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 SIGNATUREALGOINFORM 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:

对于此方案，请完成这些步骤为了安全地更新CTL文件，不用需要使用步骤丢失的eTokens，在CTL文件手工的删除结果从所有的IP电话：

1. 获取对CUCM发行商节点CLI的管理访问。

2. 输入文件删除tftp CTLFile.tlv命令到发行商节点CLI为了删除CTL文件：

```
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. 打开**SafeNet**让CTL客户端安装Microsoft Windows计算机的**验证客户端**(自动地安装与CTL客户端)：

4. 在SafeNet验证客户端，请导航对**先进的视图**：

5. 插入第一硬件USB eToken。

6. 选择证书在**用户证书**文件夹下并且导出它到在PC的文件夹。当提示输入密码，请使用**Cisco123**默认密码：

7. 重复第二硬件USB eToken的这些步骤，以便两证书导出对PC:

8. 登录Cisco Unified操作系统(OS)管理并且导航对**安全 > Certificate Management > 加载证书**：

9. 加载证书页然后出版。从证书目的下拉菜单选择电话SAST**托拉斯**并且选择您从第一eToken导出的证书：

10. 完成上一个步骤为了上传您从第二eToken导出的证书：

11. 运行CTL客户端，提供CUCM发行商节点的IP地址/主机名，并且输入CCM管理员凭证：

12. 因为集群已经在混合模式，但是CTL文件在发行商节点不存在，此警告消息出现(请点击OK键为了忽略它)：

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

13. 从CTL客户端，请点击**更新CTL File**单选按钮，其次然后单击：

14. 插入第一安全标记并且点击OK键：

15. 在安全令牌的详细信息显示后，请单击**添加**：

16. 一旦CTL文件的内容出现，请单击**添加令牌**为了添加第二USB eToken：

17. 在安全令牌的详细信息出现后，请单击**添加**：

18. 在CTL文件的内容出现后，请点击**芬通社**。当提示输入密码，请输入**Cisco123**：

19. 当的CUCM服务器列表CTL文件存在时发表，请点击**完成**：

20. 重新启动TFTP和CallManager服务在管理这些服务的所有节点在集群。

21. 重新启动所有IP电话，以便他们能从CUCM TFTP服务得到CTL文件的新版本。

22. 为了验证CTL文件的内容，请输入**显示ctl**命令到CLI。在CTL文件中您能看到从两个的证书USB eTokens (他们中的一个用于为了签署CTL文件)。以下为示例输出：

```
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. 在IP电话侧，您能验证那，在IP电话重新启动后，他们下载更新CTL文件版本(MD5校验和配比，当与从CUCM的输出比较)：

此更改是可能的，因为您以前导出并且上传eToken证书到CUCM证书托拉斯存储，并且IP电话能验证使用为了签署CTL文件托拉斯验证服务的此未知证书(TV)在CUCM的该运行。此日志snippit说明IP电话如何与CUCM TV联系以请求验证未知eToken证书，上传作为电话SAST托拉斯和委托：

```
//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

```

Tokenless CTL解决方案的证书重新生成

当使用时，此部分描述如何重新生成CUCM集群安全证书Tokenless CTL解决方案。

在CUCM维护过程中，CUCM发行商节点CallManager证书有时更改。这能发生的方案包括主机名、更改域或者完全证书重新生成更改(由于关闭证书到期到期日)。

在CTL文件更新后，比在CTL文件存在IP电话安装的那些签字与一不同的身份验证。通常，这新建的CTL文件没有接受;然而，在IP电话查找使用为了签署CTL文件的未知证书后，它与在CUCM的TV服务联系。

Note:TV服务器列表在IP电话配置文件和被映射在从IP电话设备池> Callmanager组的CUCM服务器。

在TV服务器的成功验证，IP电话更新其有新版本的CTL文件。这些事件在这种情况下发生：

1. CTL文件存在CUCM和在IP电话。CUCM发行商节点的CCM+TFT (服务器)证书用于为了签署CTL文件：

```

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. CallManager.pem文件(CCM+TFTP证书)被重新生成和您能看到证书的序列号更改：

3. 使用情况ctl更新CTLFile命令被加入到CLI为了更新CTL文件：

```
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. TV服务更新其有新的CTL文件详细信息的证书缓存：

```
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. 当您查看CTL文件目录时，您能看到文件签字与发行商节点的新的CallManager服务器证书：

```
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. 从Unified维护性页，TFTP和Cisco CallManager服务在管理这些服务的所有被重新启动在集群的节点。
7. IP电话重新启动，并且他们联系TV服务器为了验证当前使用为了签署CTL文件的新版本的未知证书：

```
// 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. 最后，在IP电话，您能验证CTL文件更新与新版本，并且新的CTL文件的MD5校验和配比与那CUCM：