CUCM、IP Phone、およびCUBE間のSIP TLSおよびSRTPのエンタープライズCA（サードパーティCA）署名付き証明書の設定とトラブルシューティング

内容

概要

このドキュメントでは、エンタープライズ認証局(CA)（3番目の認証局）を使用して、Cisco Unified Communications Manager(CUCM)、IP電話、およびCisco Unified Border Element(CUBE)間のSession Initiation Protocol(SIP)Transport Layer Security(TLS)およびSecure Real-Transport Protocol(SRTP)のののの設定例についてについてを説明しますユーザCA)署名付き証明書を使用し、共通のエンタープライズCAを使用して、IP電話、CUCM、ゲートウェイ、CUBEなどのシスココミュニケーションデバイスを含むすべてのネットワークコンポーネントの証明書に署名します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- エンタープライズCAサーバが設定されている
- CUCMクラスタは混合モードで設定され、IP Phoneはセキュアモード（暗号化）で登録されます
- CUBEの基本voice service voip dial-peerおよび設定が行われます

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- Windows 2008 Server – 認証局
- CUCM 10.5
- CUBE - Cisco IOS® 15.3(3) M3を搭載した3925E
- CIPC

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。対象のネットワークが実稼働中である場合には、どのようなコマンドについても、その潜在的な影響について確実に理解しておく必要があります。

背景説明

CUBEでのセキュアな音声通信は2つの部分に分割できます

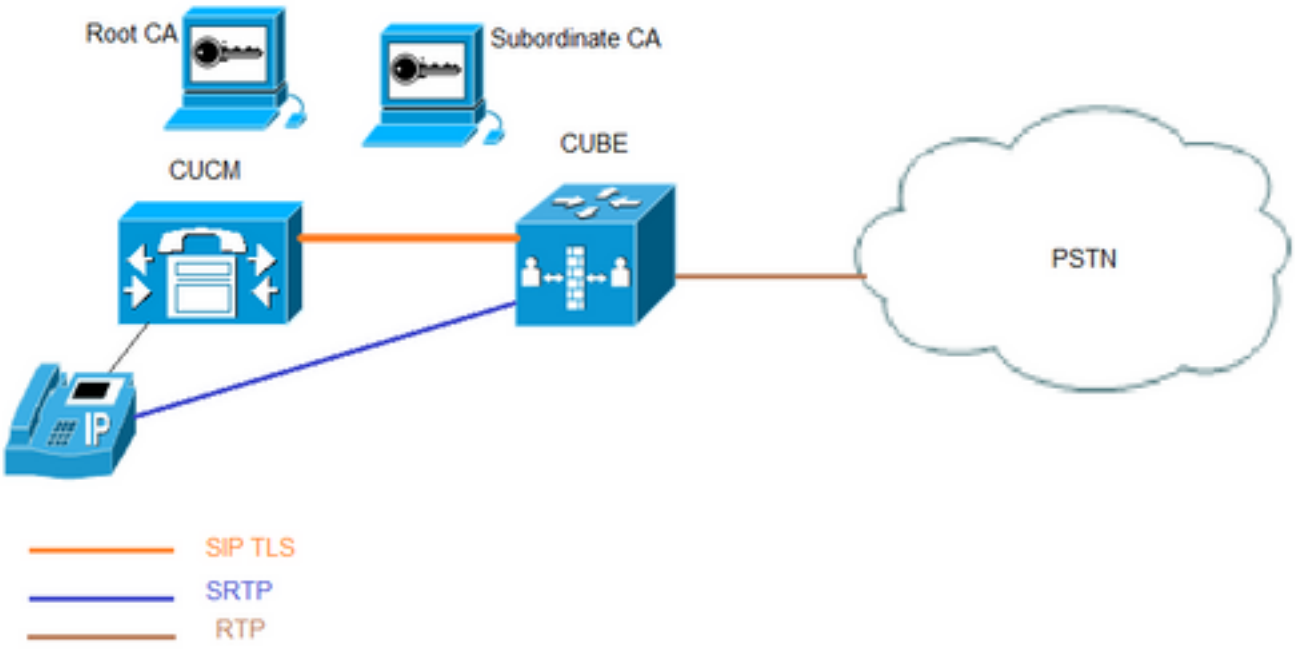
- セキュアシグナリング：CUBEはTLSを使用してSIP経由のシグナリングを保護し、H.323経由のシグナリングを保護するためにInternet Protocol Security(IPSec)を使用します
- セキュアメディア – Secure Real-Time Transport Protocol(SRTP)

CUCM Certificate Authority Proxy Function(CAPF)は、電話機にローカルで有効な証明書(LSC)を提供します。したがって、CAPFが外部CAによって署名されると、電話機の下位CAとして機能します。

CA署名付きCAPFを取得する方法については、次を参照してください。

設定

ネットワーク図



この設定では、ルートCAと1つの下位CAが使用されます。すべてのCUCMおよびCUBE証明書は、下位CAによって署名されます。

CUBEの設定

RSAキーペアを生成します。

この手順は、秘密鍵と公開鍵を生成します。

この例では、CUBEは単なるラベルですが、これは何でもかまいません。

```
CUBE-2(config)#crypto key generate rsa general-keys label CUBE modulus 2048
The name for the keys will be: CUBE

% The key modulus size is 2048 bits
% Generating 2048 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 12 seconds)

CUBE-2(config)#
```

2.下位CAおよびルートCAのトラストポイントを作成し、下位CAトラストポイントをSIP TLS通信に使用します。

この例では、下位CAのトラストポイント名はSUBCA1で、ルートCAのトラストポイント名はROOTです。

```
enrollment terminal pem allow manual cut-and-paste certificate enrollment. pem keyword is used
to issue certificate requests or receive issued certificates in PEM-formatted files through the
console terminal.
```

このステップで使用されるサブジェクト名はCUCM SIPトランク セキュリティ プロファイルの[X.509のサブジェクト名と一致する必要があります。ベストプラクティスは、（ドメイン名が有効な場合）ドメイン名でhost-nameを使用することです。

手順1.で作成された関連のRSAキー ペア。

```
crypto pki trustpoint SUBCA1
enrollment terminal pem
serial-number none
ip-address none
subject-name CN=CUBE-2
revocation-check none
rsakeypair CUBE

crypto pki trustpoint ROOT
enrollment terminal
revocation-check none
```

3. CUBE証明書署名要求(CSR)を生成します。

crypto pki enrollコマンドは、署名付き証明書を取得するためにエンタープライズCAに提供されるCSRを生成します。

```
CUBE-2(config)#crypto pki enroll SUBCA1
% Start certificate enrollment ..

% The subject name in the certificate will include: CN=CUBE-2
% The subject name in the certificate will include: CUBE-2
Display Certificate Request to terminal? [yes/no]: yes
Certificate Request follows:
```

```
-----BEGIN CERTIFICATE REQUEST-----
MIICjjCCAXYCAQAwKDEPMA0GA1UEAxMGQ1VCRS0yMRUwEwYJKoZIhvcNAQkCFgZD
VUJFLTIwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDAmVvufevAg1ip
Kn8FhWjFlNNUFMqkgh2Cr1IMV+ovR2HyPTFwgr0XDhZHMSsnBw67Ttze3Ebxxoau
cBQcIASZ4hdTSIgjxG+9YQacLm9MXpfxHp5kcICzSfSllrTexArTQglW8+rErYpk
2THN1S0PC4cRlBwoUCgB/+KCDkjJkUy8eCX+Gmd+6ehRKEQ5HdFHEfUr5hc/7/pB
liHietNKSxYEOr9TVZPiRJrtpUPMRMZElRUm7GoxBrCWIXVdvEAGC0Xqd1ZVLlTz
z2sQQDqvJ9fMN6fngKv2ePr+f5qejWVzGO0DFVQs0y5x+Yl+pHbsdV1hSSnPpJk6
TaaBmX83AgMBAAGgITAfBgkqhkiG9w0BCQ4xEjAQMA4GA1UdDwEB/wQEAwIFoDAN
BgkqhkiG9w0BAQUFAAOCAQEArWMJbdhlU8VfaF1cMJIbr569BZT+tIjQOz3OqNGQ
QpzHwclLoaKuC5pc/u0hw14MGS6Z440Iw4zK2/5bb/KL47r8H3d7T7PYMfK6lAzK
sU9Kf96zTvHNWl9wXImB5blJfRLXnFWXNsVEF4FjU74plxJL7siaa5e86eNy9deN
20iKjvP8o4MgWewILrD01YZMDMDS1Uy82kWI6hvXG5+xBT5A1lo2xCj1S9y6/D4d
f0ilDZvaQk+7jjBCzLv5hET+1neoQBw52e7RWU8s2biQw+7TEAdO8NytF3q/mA/x
bUKw5wT4pgGUJcDAWej3ZLqP91g5yyd9MiCdCRY+3mLccQ==
-----END CERTIFICATE REQUEST-----

---End - This line not part of the certificate request---

Redisplay enrollment request? [yes/no]: no
CUBE-2(config)#
```

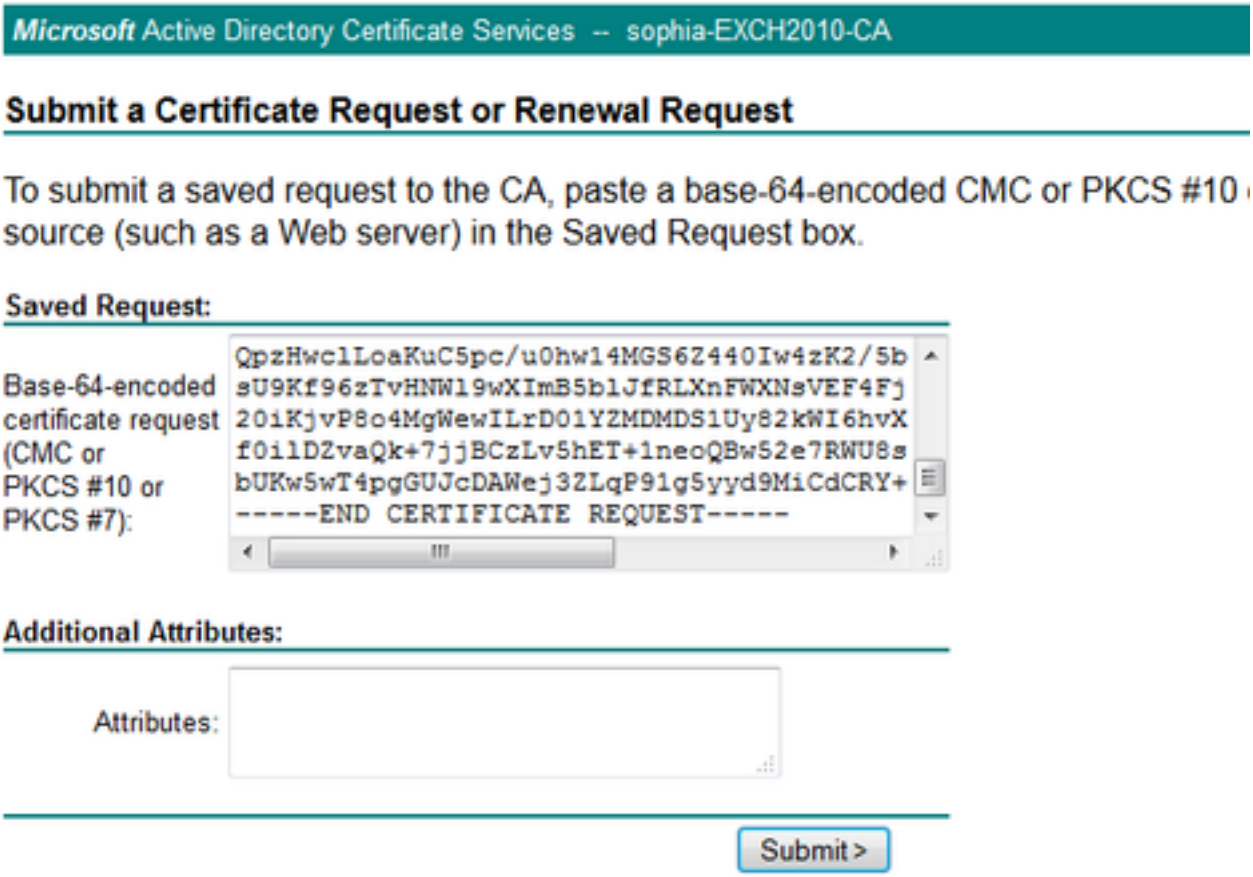
BEGIN CERTIFICATE REQUESTからEND CERTIFICATE REQUESTの間の出力をコピーし、メモ帳ファイルに保存します。

CUBE CSRには次のキー属性があります。

```
 Attributes:
Requested Extensions:
X509v3 Key Usage: critical
Digital Signature, Key Encipherment
```

4. CA証明書のルートCA、CA証明書、および署名済みCUBE証明書を下位CAから取得します。

署名付きCUBE証明書を取得するには、手順3で生成されたCSRを使用します。イメージはMicrosoft CA Webサーバからのものです。



5.ルートCAと下位CAのCA証明書をインポートします。

メモ帳で証明書を開き、BEGIN CERTIFICATE REQUESTからEND CERTIFICATE REQUESTに内容をコピーアンドペーストします。

```
CUBE-2(config)#crypto pki authenticate SUBCA1

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIFhDCCBGygAwIBAgIKYZVFyQAAAAAAFjANBgkqhkiG9w0BAQUFADBQMRIwEAYK
CZImiZPyLGQBGRYCbGkxFjAUBgoJkiaJk/IsZAEZFgZzb3BoaWExIjAgBgNVBAMT
GXNvcGhpYS1XSU4tM1MxOEpDM0xNMkEtQ0EwHhcNMTQwOTI1MDAwNzU2WhcNMTYw
OTI1MDAxNzU2WjBJMRIwEAYKCZImiZPyLGQBGRYCbGkxFjAUBgoJkiaJk/IsZAEZ
FgZzb3BoaWExGzAZBgNVBAMTEnNvcGhpYS1FWENIMjAxMC1DQTCCASIwDQYJKoZI
hvcNAQEBBQADggEPADCCAQoCggEBAJK+Nmz4rieYfr9gH3ISTuYz3TWpafpjDJ7l
7kIwwwC28TvjFl5vrKEiaPyFzxL5TEHaWQ9YAo/WMdtuyF7aB+pLJ1soKcZxtrGv
gTMtuphcJ5Fpd4368lR8ZXJiAT/Dz+Nsh4PC9GUUKQeycyRDeOBz08vL5pLj/W99
b8UMUlVOqBu4e1ZwxWPMFxB7zOeYsCfXMnGFUlp3HFdWZczgK3ldNO9I0X+p70UP
R0CQpMEQxuheqv9kazIIJKfNH8N0qO8IHl76Y32vUzLg3uvZgqWG6hGch/gjm4L/
1KmdZTNSH8H7Kf6vG6PNWrXWwLNkhrWaYEryHelIshEj7ZUeB8sCAwEAAaOCAmUw
ggJhMBIGCSsGAQQBgjcVAQQFAgMBAAEwIwYJKwYBBAGCNxUCBBYEFLnnd8HnCfKE
isPgI58Oog/LqwVSMB0GA1UdDgQWBBSsdYJZIU9IXyGm9aL67+8uDhM/EzAZBgkr
BgEEAYI3FAIEDB4KAFMAdQBiAEMAQTAOBgNVHQ8BAf8EBAMCAYYwDwYDVR0TAQH/
BAUwAwEB/zAfBgNVHSMEGDAWgBTvo1P6OP4LXm9RDv5MbIMk8jnOfDCB3QYDVR0f
BIHVMIHSMIHPoIHMoIHJhoHGbGRhcDovLy9DTj1zb3BoaWEtV0lOLTNTMThKQzNM
TTJBLUNBLENOPVdJTi0zUzE4SkMzTE0yQSxDTj1DRFAsQ049UHVibGljJTIwS2V5
JTIwU2VydmljZXMsQ049U2VydmljZXMsQ049Q29uZmlndXJhdGlvbixEQz1zb3Bo
aWEsREM9bGk/Y2VydGlmaWNhdGVSZXZvY2F0aW9uTGlzdD9iYXNlP29iamVjdENs
YXNzPWNSTERpc3RyaWJ1dGlvblBvaW50MIHJBggrBgEFBQcBAQSBvDCBuTCBtgYI
KwYBBQUHMAKGgalsZGFwOi8vL0NOPXNvcGhpYS1XSU4tM1MxOEpDM0xNMkEtQ0Es
```

```
Q049QUlBLENOPVB1YmxpYyUyMEtleSUyMFNlcnZpY2VzLENOPVNlcnZpY2VzLENO
PUNvbmZpZ3VyYXRpb24sREM9c29waGlhLERDPWxpP2NBQ2VydGlmaWNhdGU/YmFz
ZT9vYmplY3RDbGFzcz1jZXJ0aW9uQXV0aG9yaXR5MA0GCSqGSIb3DQEB
BQUAA4IBAQBj/+rX+9NJiSZq1YwQXkLq6+LUh7OkCoeCHHfBGUaS+gvbYQ5OVwJI
TlPTj4YNh62A6pUXplo8mdxKxOmZeRLTYgf9Q/SiOY+qoxJ5zNliSqlRU4E02sRz
wrzfaQpLGgyHXsyK1ABOGRgGqqWqZ7oXoKMRNmO+eu3NzBs4AVAAfL8UhFCv4IVx
/t6qIHY6YkNMVByjZ3MdFmohepN5CHZUHIvrOv9eAiv6+Vaan2nTeynyy7WnEv7P
+5L2kEFOSfnL4Zt2tEMqc5WyX6yjxDWmII0DTSyRshmxAoYlo3EJHwW+fIocdmIS
hgWDzioZ70SM9mJqNReHMC1jL3FD2nge
-----END CERTIFICATE-----

Trustpoint 'SUBCA1' is a subordinate CA and holds a non self signed cert
Certificate has the following attributes:
Fingerprint MD5: C420B7BB 88A2545F E26B0875 37D9EB45
Fingerprint SHA1: 110AF87E 53E6D1C2 19404BA5 0149C5CA 2CF2BE1C

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported

CUBE-2(config)#
CUBE-2(config)#crypto pki authenticate ROOT

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIDezCCAmOgAwIBAgIQMVF/OWq+ELxFC2IdUGvd2jANBgkqhkiG9w0BAQUFADBQ
MRIwEAYKCZImiZPyLGQBGRYCbGkxFjAUBgoJkiaJk/IsZAEZFgZzb3BoaWExIjAg
BgNVBAMTGXNvcGhpYS1XSU4tM1MxOEpDM0xNMkEtQ0EwHhcNMTQwOTEzMjMzODA2
WhcNMTkwOTEzMjM0ODA1WjBQMRIwEAYKCZImiZPyLGQBGRYCbGkxFjAUBgoJkiaJ
k/IsZAEZFgZzb3BoaWExIjAgBgNVBAMTGXNvcGhpYS1XSU4tM1MxOEpDM0xNMkEt
Q0EwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC4aywr1oOpTdTrM8Ya
R3RkcahbbhR3q7P1luTDUDNM5Pi6P8z3MckfjB/yy6SWr1Qnddhyv MG6IGNtVxJ4
eyw0c7jbArXWOemGLOt454A0mCfcbwMhjQBycg9SM1r1Umzad7kOCzj/rD6hMbC4
jXpg6uU8g7eB3LzN1XF93DHjxYCBKMIeG45pqmsOc3mUj1CbCtnYXgno+mfhNzhR
HSth02z4XlGm99v46j/PqGjNRq4WKCwDc45SG3QjJDqDxnRJPKtRdNva66UJfDJp
4YMXQxOSkKMtDEDhH/Eic7CrJ3EywpUpMZAmqh4bmQ7Vo2pnRTbYdaAv/+yr8sMj
+FU3AgMBAAGjUTBPMAsGA1UdDwQEAwIBhjAPBgNVHRMBAf8EBTADAQH/MB0GA1Ud
DgQWBBTvo1P6OP4LXm9RDv5MbIMk8jnOfDAQBgkrBgEEAYI3FQEEAwIBADANBgkq
hkiG9w0BAQUFAAOCAQEAmd7hJ2EEUmuMZrc/qtSJ223loJlpKEPMVi7CrodtWSgu
5mNt1XsgxijYMqD5gJe1oq5dmv7efYvOvI2WTCXfwOBJ0on8tgLFwp1+SUJWs95m
OXTyoS9krsI2G2kQkjQWniMqPdNxpmJ3C4WvQLPLwtEOSRZRBvsKy6lczrgrV2mZ
kx12n5YGrGcXSblPPUddlJep1l8U+AQC8wkSzfJu0yHJwoH+lrIfgqKUee4x7z6s
SCaGddCYr3OK/3Wzs/WjSO2UETvNL3NEtWHDc2t4Y7mmIMSDvGjHZUgGZotwc9kt
9f2dZA0rtgBq4IDtpxkR3CQaauB7wUCpzemHzf+z9Q==
-----END CERTIFICATE-----

Certificate has the following attributes:
Fingerprint MD5: 511E1008 6D315E03 4B748601 7EE1A0E5
Fingerprint SHA1: 8C35D9FA 8F7A00AC 0AA2FCA8 AAC22D5F D08790BB

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported

CUBE-2(config)#
```

6. CUBE署名付き証明書をインポートします。

メモ帳で証明書を開き、BEGIN CERTIFICATE REQUESTからEND CERTIFICATE REQUESTに内容をコピーアンドペーストします。

```
CUBE-2(config)#crypto pki import SUBCA1 certificate

Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----
MIIEAjCCAuqgAwIBAgIKQZZrHQABAAAAEzANBgkqhkiG9w0BAQUFADBJMRIwEAYK
CZImiZPyLGQBGRYCbGkxFjAUBgoJkiaJk/IsZAEZFgZzb3BoaWExGzAZBgNVBAMT
EnNvcGhpYS1FWENIMjAxMC1DQTAeFw0xNTA0MDEwMDEzNDFaFw0xNjA0MDEwMDIz
NDFaMBExDzANBgNVBAMTBkNVQkUtMjCCASIwDQYJKoZIhvcNAQEBBQADggEPADCC
AQoCggEBAMCZW+5968CDWKkqfwWFaMWU01QUyqSCHYKvUgxX6i9HYfI9MXCCvRcO
FkcxKycHDrtO3N7cRvHGhq5wFBwgBJniF1NIiCPEb71hBpwub0xel/EenmRwgLNJ
9KWWtN7ECtNCCVbz6sStimTZMc3VLQ8LhxGUHChQKAH/4oIOSMmRTLx4Jf4aZ37p
6FEoRDkd0UcR9SvmFz/v+kGWIeJ600pLFgQ6v1NVk+JEmu2lQ8xExkSVFSbsajEG
sJYhdV28QAYLRep3VlUuVPPPaxBAOq8n18w3p+eAq/Z4+v5/mp6NZXMY7QMVVCzT
LnH5iX6kdux1XWFJKc+kmTpNpoGZfzcCAwEAAaOCASIwggEeMA4GA1UdDwEB/wQE
AwIFoDAdBgNVHQ4EFgQU9PbHMHSkYrjJ2+/+hSSMEoma0QIwHwYDVR0jBBgwFoAU
rHWCWSFPSF8hpvWi+u/vLg4TPxMwTwYDVR0fBEgwRjBEoEKgQIY+ZmlsZTovL0VY
Q0gyMDEwLnNvcGhpYS5saS9DZXJ0RW5yb2xsL3NvcGhpYS1FWENIMjAxMC1DQSgx
KS5jcmwwbQYIKwYBBQUHAQEEYTBfMF0GCCsGAQUFBzAChlFmaWxlOi8vRVhDSDIw
MTAuc29waGlhLmxpL0NlcnRFbnJvbGwvRVhDSDIwMTAuc29waGlhLmxpX3NvcGhp
YS1FWENIMjAxMC1DQSgxKS5jcnQwDAYDVR0TAQH/BAIwADANBgkqhkiG9w0BAQUF
AAOCAQEAe7EAoXKIAij4vxZuxROOFOfsmjcojU31ac5nrLCbq/FyW7eNblphL0NI
Dt/DlfZ5WK2q3Di+/UL1lDt3KYt9NZ1dLpmccnipbbNZ5LXLoHDkLNqt3qtLfKjv
J6GnnWCxLM18lxm1DzZT8VQtiQk5XZ8SC78hbTFtPxGZvfX70v22hekkOL1Dqw4h
/3mtaqxfnslB/J3Fgps1och45BndGiMAWavzRjjOKQaVLgVRvVrPIy3ZKDBaUleR
gsy5uODVSrhwMo3z84r+f03k4QarecgwZE+KfXoTpTAfhiCbLKw0ZyRMXXzWqNfl
iotEQbs52neCwXNwV24aOCChQMw2xw==
-----END CERTIFICATE-----

% Router Certificate successfully imported

CUBE-2(config)#
```

7. TCP TLSをトランスポートプロトコルとして設定します。

これは、グローバルまたはダイヤルピアレベルで実行できます。

```
 voice service voip
sip
session transport tcp tls
```

8. sip-uaにトラストポイントを割り当てます。このトラストポイントは、CUBEとCUCM間のすべてのsipシグナリングに使用されます。

```
 sip-ua
crypto signaling remote-addr <cucm pub ip address> 255.255.255.255 trustpoint SUBCA1
crypto signaling remote-addr <cucm sub ip address> 255.255.255.255 trustpoint SUBCA1
```

または、cubeからのすべてのsipシグナリングに対してデフォルトのトラストポイントを設定できます。

```
 sip-ua
crypto signaling default trustpoint SUBCA1
```

9. SRTPを有効にします。

これは、グローバルまたはダイヤルピアレベルで実行できます。

```
Voice service voip
srtp fallback
```

10. SRTPおよびReal-time Transport Protocol(RTP)インターネットワーキングには、セキュアなトランスコーダが必要です。

Cisco IOS®バージョンが15.2.2T(CUBE 9.0)以降の場合、ローカルトランスコーディングインターフェイス(LTI)トランスコーダを設定して、設定を最小限に抑えることができます。

LTIトランスコーダでは、SRTP-RTPコールに対してPublic Key Infrastructure(PKI)トラストポイント設定は必要ありません。

```
 dspfarm profile 1 transcode universal security
codec g711ulaw
codec g711alaw
codec g729ar8
codec g729abr8
maximum sessions 10
associate application CUBE
```

Cisco IOS®が15.2.2T未満の場合は、SCCPトランスコーダを設定します。

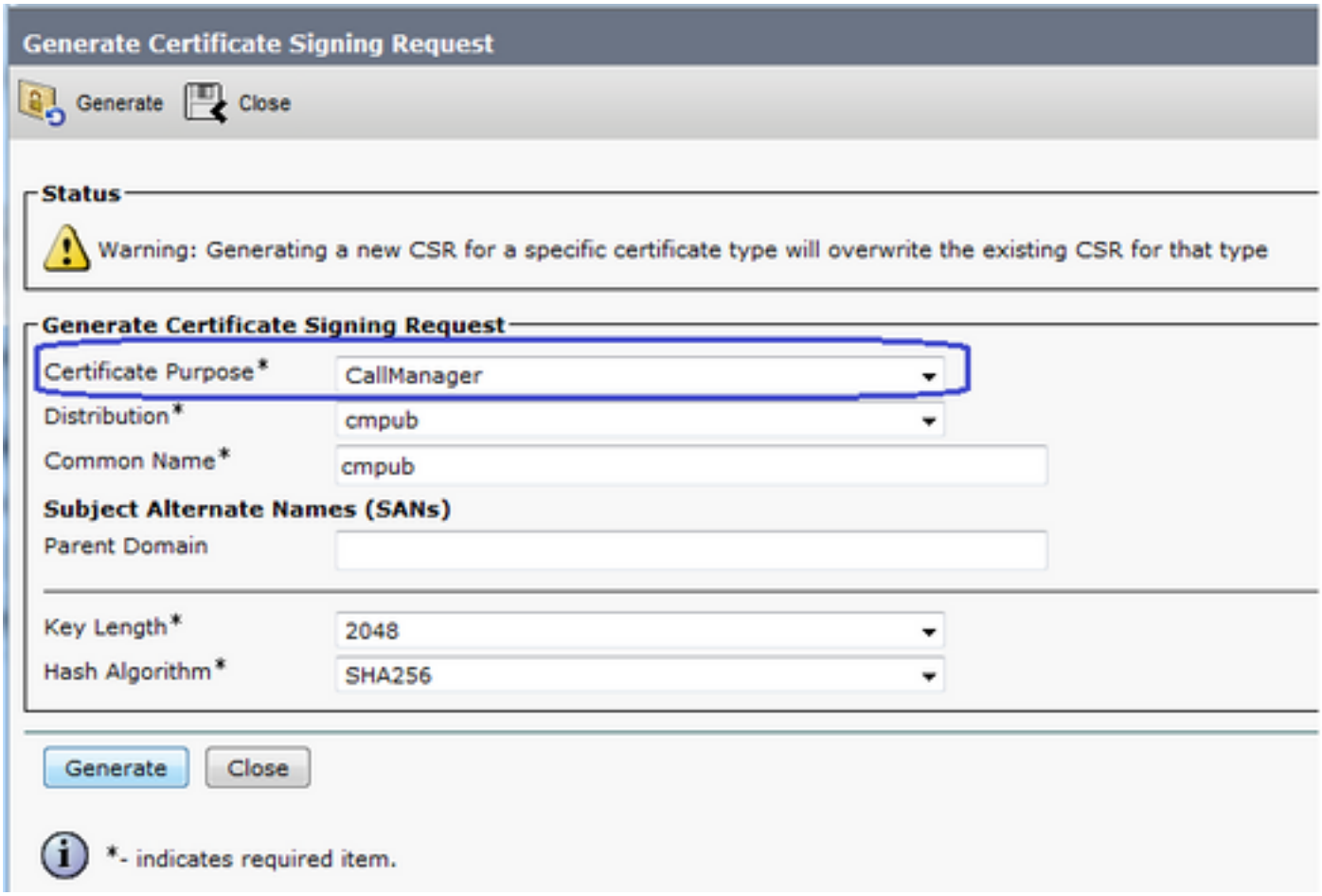
SCCPトランスコーダはシグナリングにトラストポイントを必要としますが、トランスコーダをホストするために同じルータを使用する場合、CUBEとトランスコーダに同じトラストポイント(SUBCA1)を使用できます。

```
sccp local GigabitEthernet0/2
sccp ccm 10.106.95.153 identifier 1 priority 1 version 7.0
sccp
!
sccp ccm group 1
bind interface GigabitEthernet0/0
associate ccm 1 priority 1
associate profile 2 register secxcode
!
dspfarm profile 2 transcode universal security
trustpoint SUBCA1
codec g711ulaw
codec g711alaw
codec g729ar8
codec g729abr8
maximum sessions 10
associate application SCCP

telephony-service
secure-signaling trustpoint SUBCA1
sdspfarm units 1
sdspfarm transcode sessions 10
sdspfarm tag 1 secxcode
max-ephones 1
max-dn 1
ip source-address 10.106.95.153 port 2000
max-conferences 8 gain -6
transfer-system full-consult
```

CUCM の設定

1.すべてのCUCMノードでCallManager CSRを生成します。

図に示すように、[CM OS Administration] > [Security] > [Certificate Management] > [Generate Certificate Signing Request]に移動します。

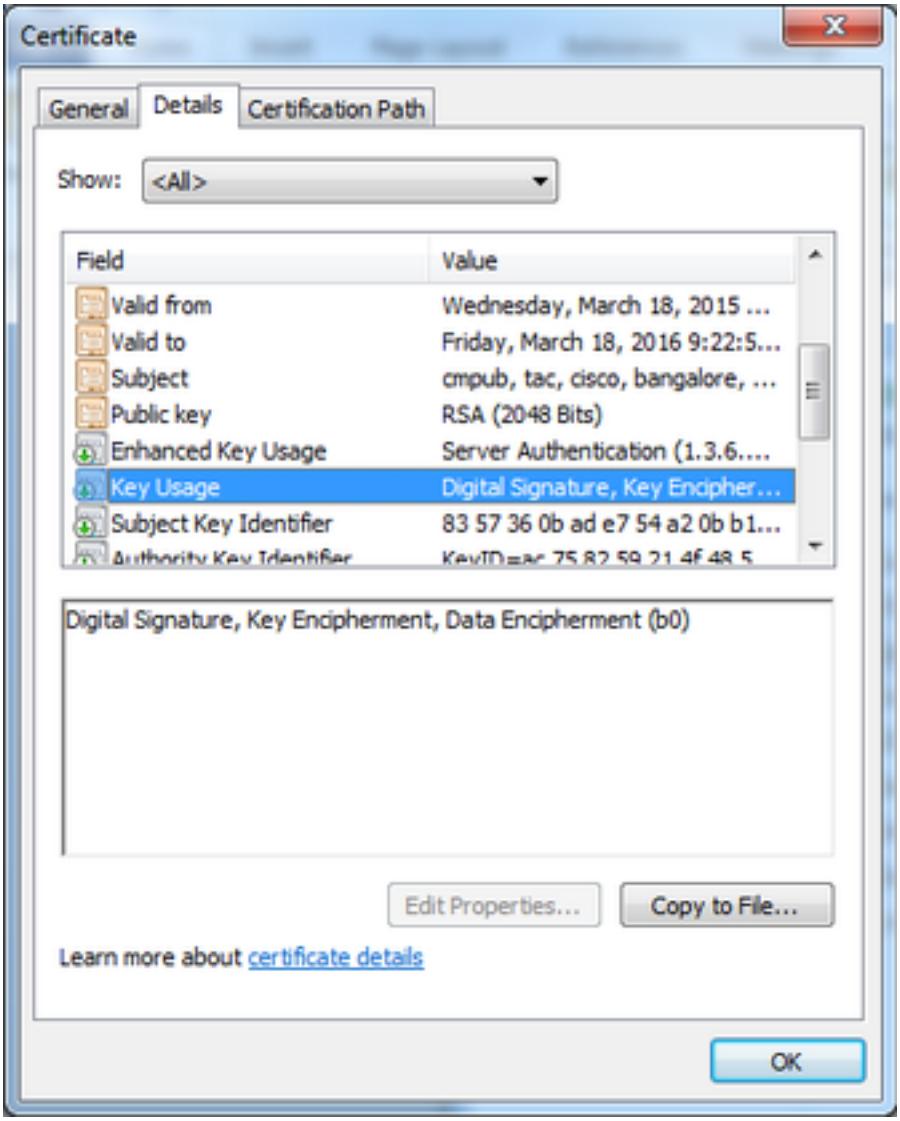


CallManager CSRには次のキー属性があります。

```
Requested Extensions:
X509v3 Extended Key Usage:
TLS Web Server Authentication, TLS Web Client Authentication, IPSec End System
X509v3 Key Usage:
Digital Signature, Key Encipherment, Data Encipherment, Key Agreement
```

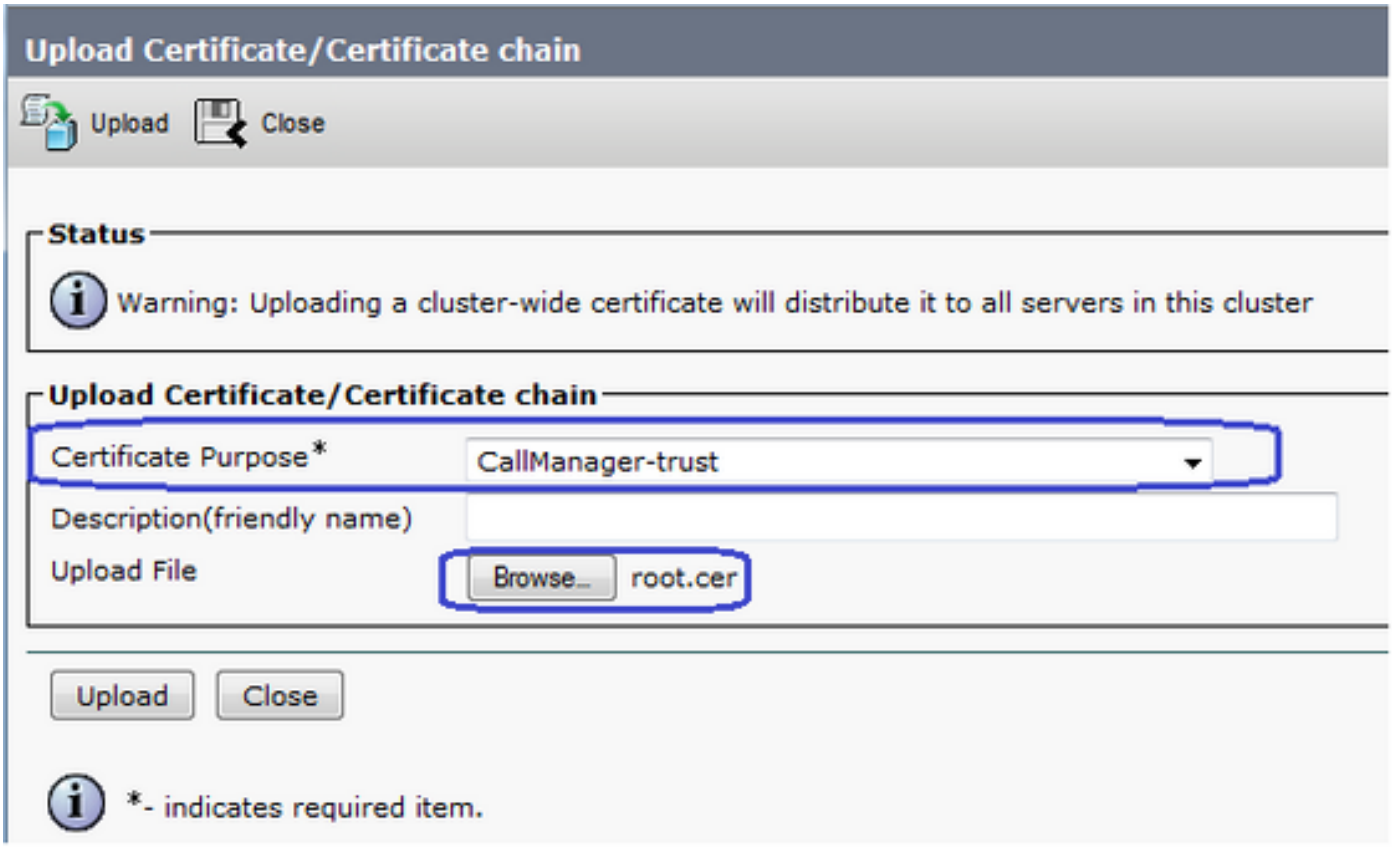
2.下位CAによって署名されたすべてのCMノードのCallManager証明書を取得します。

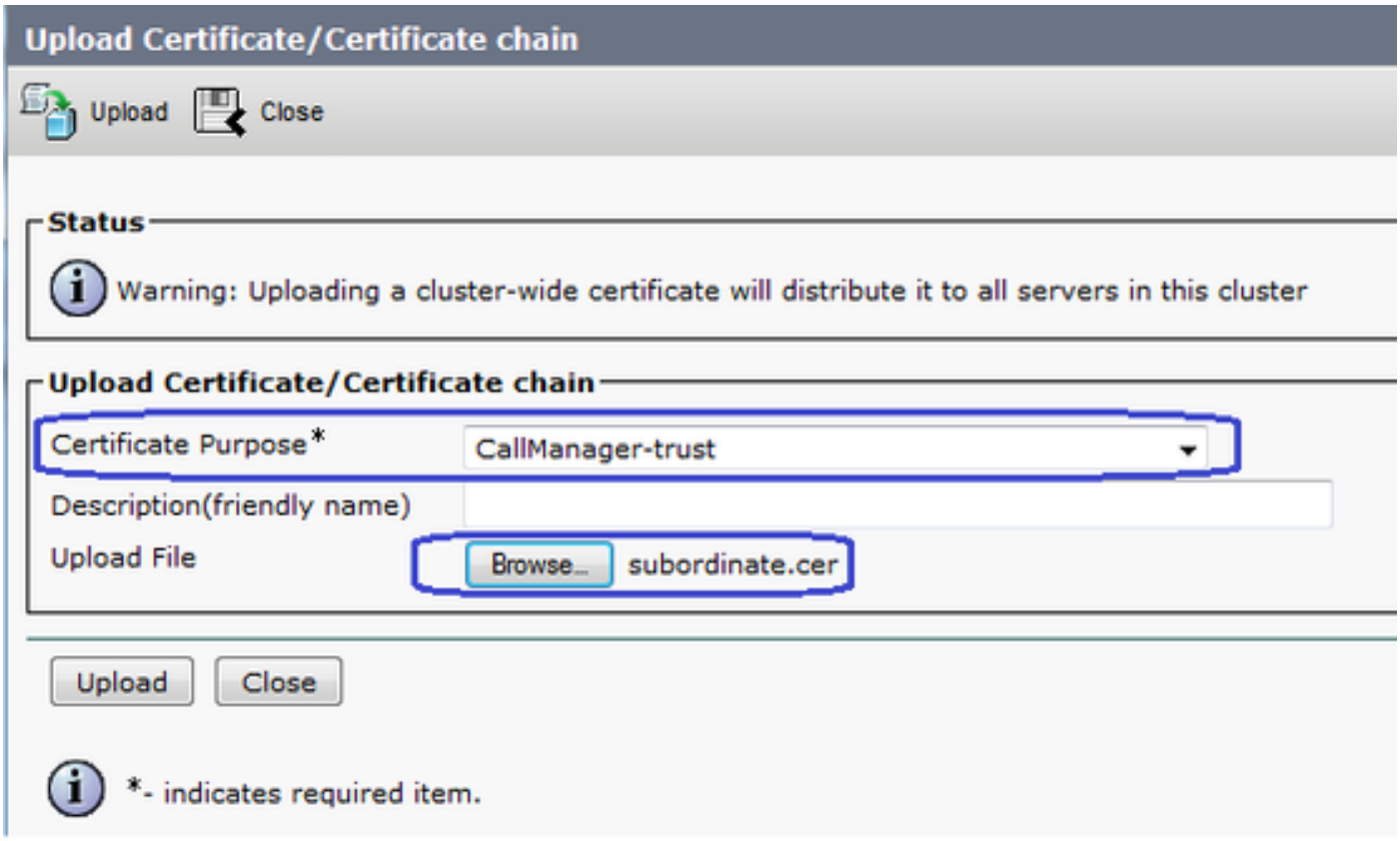
ステップ1で生成されたCSRを使用します。Webサーバ証明書テンプレートは機能し、署名付き証明書に次のキー使用属性が少なくとも含まれていることを確認します。**図に示すように、デジタル署名、キー暗号**、データ暗号化。



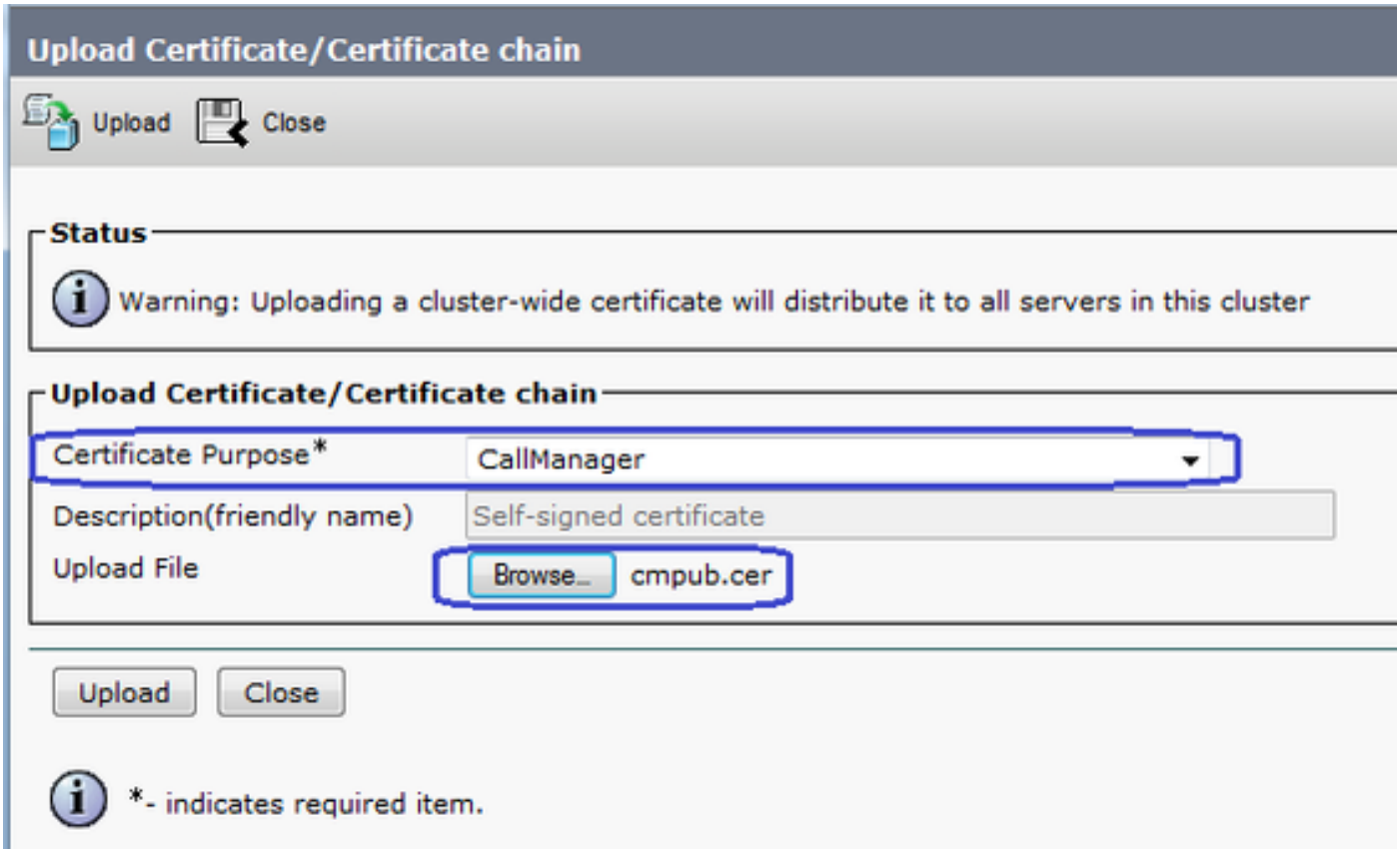
3.ルートCAおよび下位CAからCallManager-TrustとしてCA証明書をアップロードします。

図に示すように、[CM OS Administration] > [Security] > [Certificate Management] > [Upload Certificate/Certificate chain]に移動します。





4.図に示すように、CallManager署名付き証明書をCallManagerとしてアップロードします。



5.パブリッシャの証明書信頼リスト(CTL)ファイルを更新します（CLIを使用）。

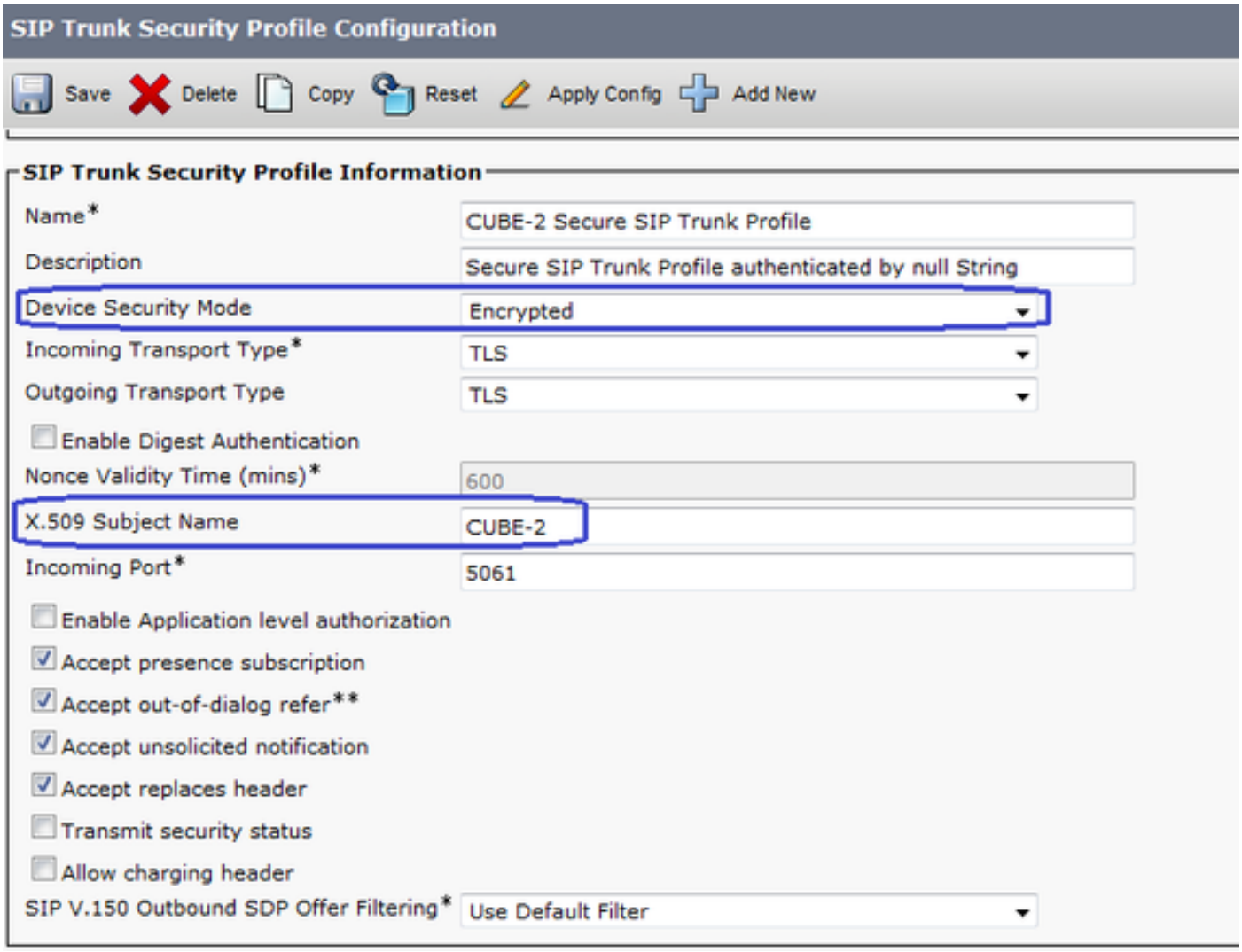
```
admin:utils ctl update CTLFile
This operation will update the CTLFile. Do you want to continue? (y/n):

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:
```

6. すべてのノードでCallManagerサービスとTFTPサービスを、パブリッシャでCAPFサービスを再起動します。

7.新しいSIPトランクセキュリティプロファイルの作成

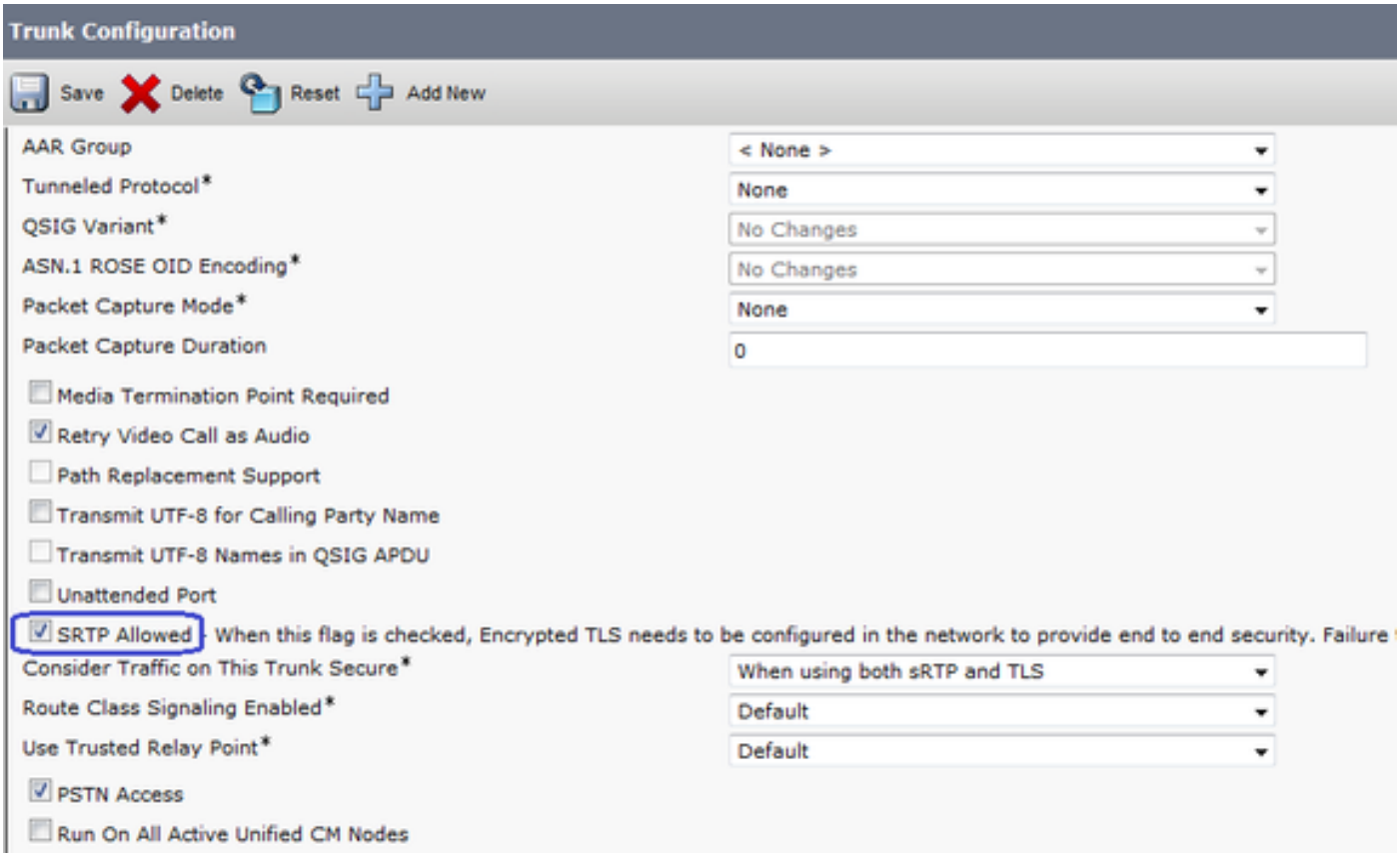
CM Administrationで、**[System] > [Security] > [SIP Trunk Security Profiles] > [Find]に移動します**。

次の図に示すように、既存の非セキュアSIPトランクプロファイルをコピーして、新しいセキュアプロファイルを作成します。

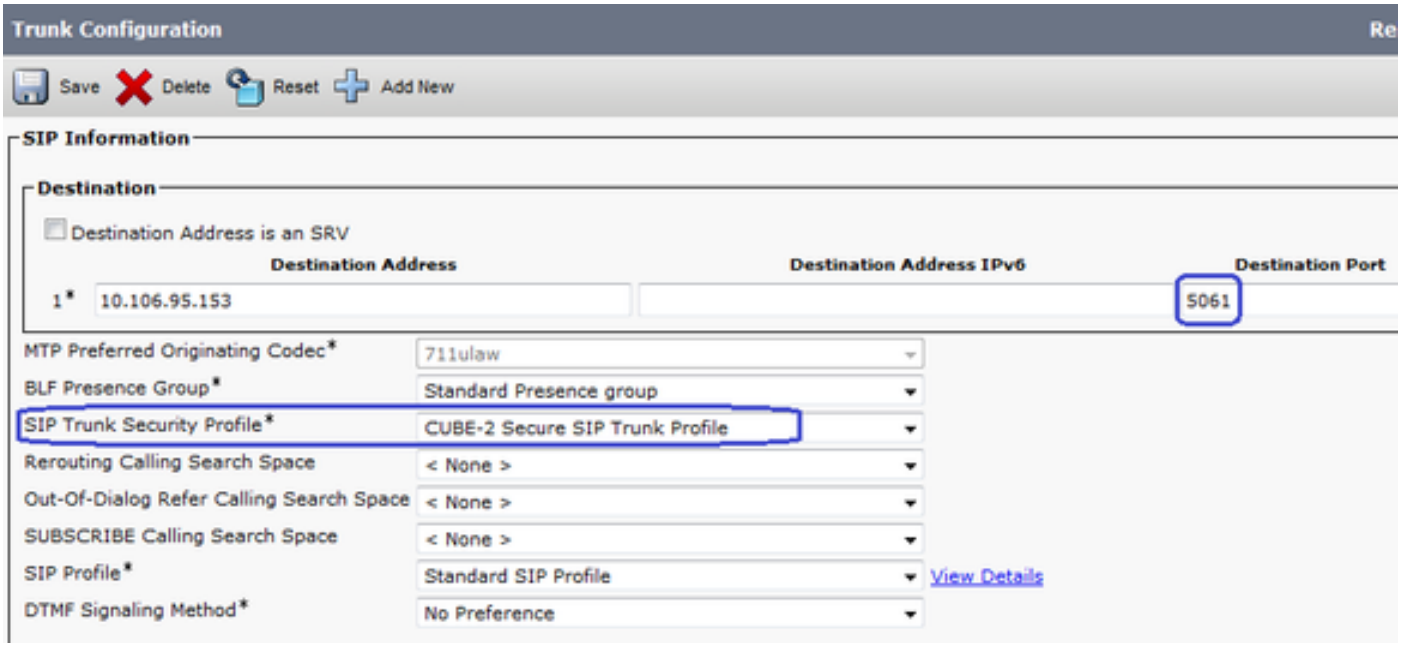


8. CUBEへのSIPトランクを作成します.

図に示**すように**、SIPトランクで[SRTP Allowed]を有効にします。



図に示すように、宛先ポート5061(TLS)を設定し、新しいセキュアSIPトランクセキュリティプロファイルをSIPトランクに適用します。



確認

ここでは、設定が正常に機能しているかどうかを確認します。

```
show sip-ua connections tcp tls detail
show call active voice brief
```

e.g.

```
Secure-CUBE#show sip-ua connections tcp tls detail
Total active connections : 2
No. of send failures : 0
No. of remote closures : 13
No. of conn. failures : 0
No. of inactive conn. ageouts : 0
TLS client handshake failures : 0
TLS server handshake failures : 0

---------Printing Detailed Connection Report---------
Note:
** Tuples with no matching socket entry
- Do 'clear sip <tcp[tls]/udp> conn t ipv4:<addr>:<port>'
to overcome this error condition
++ Tuples with mismatched address/port entry
- Do 'clear sip <tcp[tls]/udp> conn t ipv4:<addr>:<port> id <connid>'
to overcome this error condition

Remote-Agent:10.106.95.151, Connections-Count:2
Remote-Port Conn-Id Conn-State WriteQ-Size Local-Address
=========== ======= =========== =========== ===========
5061 16 Established 0 10.106.95.153
57396 17 Established 0 10.106.95.153


-------------- SIP Transport Layer Listen Sockets ---------------
Conn-Id Local-Address
=========== ============================
2 [10.106.95.153]:5061
```

show call active voice briefコマンドの出力は、LTIトランスコダが使用されている場合にキャプチャされます。

```
Telephony call-legs: 0
SIP call-legs: 2
H323 call-legs: 0
Call agent controlled call-legs: 0
SCCP call-legs: 0
Multicast call-legs: 0
Total call-legs: 2
1283 : 33 357052840ms.1 (23:57:23.929 IST Sun Feb 15 2015) +2270 pid:3 Answer 3001 active
dur 00:00:08 tx:383/61280 rx:371/59360 dscp:0 media:0 audio tos:0xB8 video tos:0x0
IP 10.106.95.132:17172 SRTP: off rtt:0ms pl:0/0ms lost:0/0/0 delay:0/0/0ms g711ulaw TextRelay:
off Transcoded: Yes
media inactive detected:n media contrl rcvd:n/a timestamp:n/a
long duration call detected:n long duration call duration:n/a timestamp:n/a
LostPacketRate:0.00 OutOfOrderRate:0.00

1283 : 34 357052840ms.2 (23:57:23.929 IST Sun Feb 15 2015) +2270 pid:1 Originate 2001 active
dur 00:00:08 tx:371/60844 rx:383/62812 dscp:0 media:0 audio tos:0xB8 video tos:0x0
IP 10.65.58.24:24584 SRTP: on rtt:0ms pl:0/0ms lost:0/0/0 delay:0/0/0ms g711ulaw TextRelay: off
Transcoded: Yes
media inactive detected:n media contrl rcvd:n/a timestamp:n/a
long duration call detected:n long duration call duration:n/a timestamp:n/a
LostPacketRate:0.00 OutOfOrderRate:0.00
```

また、Cisco IP PhoneとCUBEまたはゲートウェイの間でSRTP暗号化コールが行われると、IP Phoneにロックアイコンが表示されます。

トラブルシュート

ここでは、設定のトラブルシューティングに使用できる情報を示します。

これらのデバッグは、PKI/TLS/SIP/SRTPの問題のトラブルシューティングに役立ちます。

```
debug crypto pki{ API | callbacks | messages | scep | server | transactions | validation }
debug ssl openssl { errors | ext | msg | states }
debug srtp {api | events }
debug ccsip {messages | error | events | states | all }
debug voip ccapi inout
```