CA 서명 인증서를 사용하여 CUCM-CUBE/CUBE-SBC 간 SIP TLS 구성

목차

소개

이 문서에서는 CUCM(Cisco Unified Communication Manager)과 CA(Certificate Authority) 서명 인증서를 사용하는 Cisco CUBE(Unified Border Element) 간에 SIP TLS(Transport Layer Security)를 구성하는 방법에 대해 설명합니다.

사전 요구 사항

Cisco는 이러한 주제에 대해 알고 있는 것이 좋습니다.

- SIP 프로토콜
- 보안 인증서

요구 사항

- 날짜와 시간은 엔드포인트에서 일치해야 합니다(동일한 NTP 소스를 사용하는 것이 좋습니다).
- CUCM은 혼합 모드여야 합니다.
- TCP 연결이 필요합니다(모든 트랜짓 방화벽에서 포트 5061 열기).
- CUBE에는 보안 및 UCK9(Unified Communication K9) 라이센스가 설치되어 있어야 합니다.

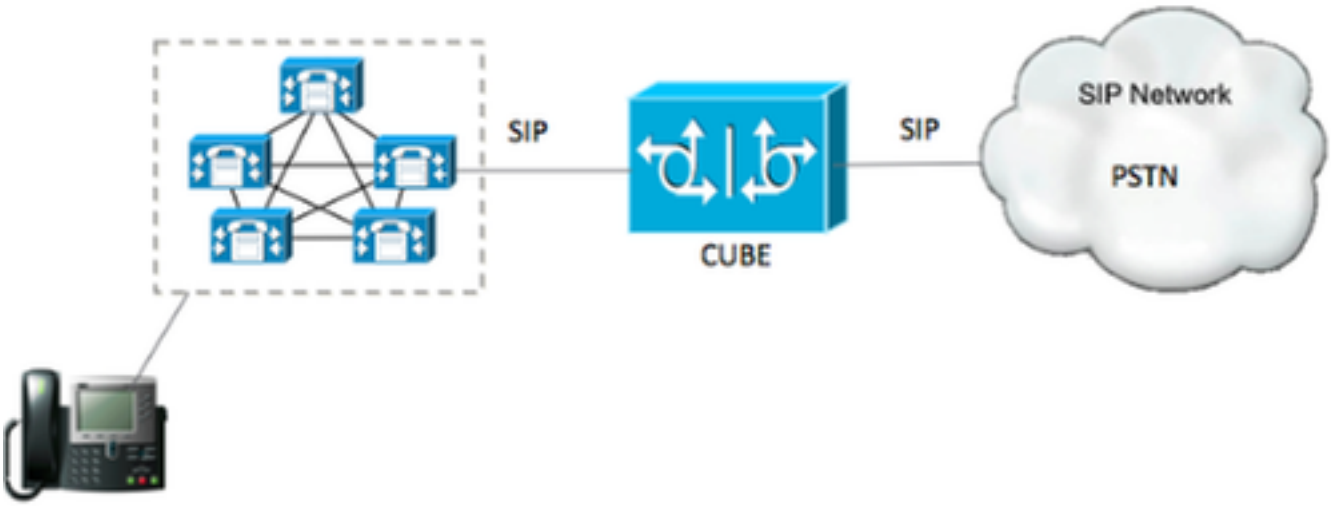
참고: Cisco IOS-XE 버전 16.10의 경우 플랫폼이 스마트 라이센싱으로 전환되었습니다.

사용되는 구성 요소

- SIP
- 인증 기관 서명 인증서
- Cisco IOS 및 IOS-XE 게이트웨이2900/3900/4300/4400/CSR1000v/ASR100X 버전:15.4+
- Cisco CUCM(Unified Communications Manager)버전:10.5 이상

구성

네트워크 다이어그램



구성

1단계. 다음 명령을 사용하여 루트 인증서의 인증서 길이와 일치하는 RSA 키를 생성합니다.

```
Crypto key generate rsa label TestRSAkey exportable modulus 2048
```

이 명령은 길이가 2048비트(최대값은 4096)인 RSA 키를 생성합니다.

2단계. 다음 명령을 사용하여 CA 서명 인증서를 보유할 신뢰 지점을 만듭니다.

```
 Crypto pki trustpoint CUBE_CA_CERT
   serial-number none
   fqdn none
   ip-address none
   subject-name cn=ISR4451-B.cisco.lab !(this has to match the router's hostname
[hostname.domain.name])
   revocation-check none
   rsakeypair TestRSAkey !(this has to match the RSA key you just created)
```

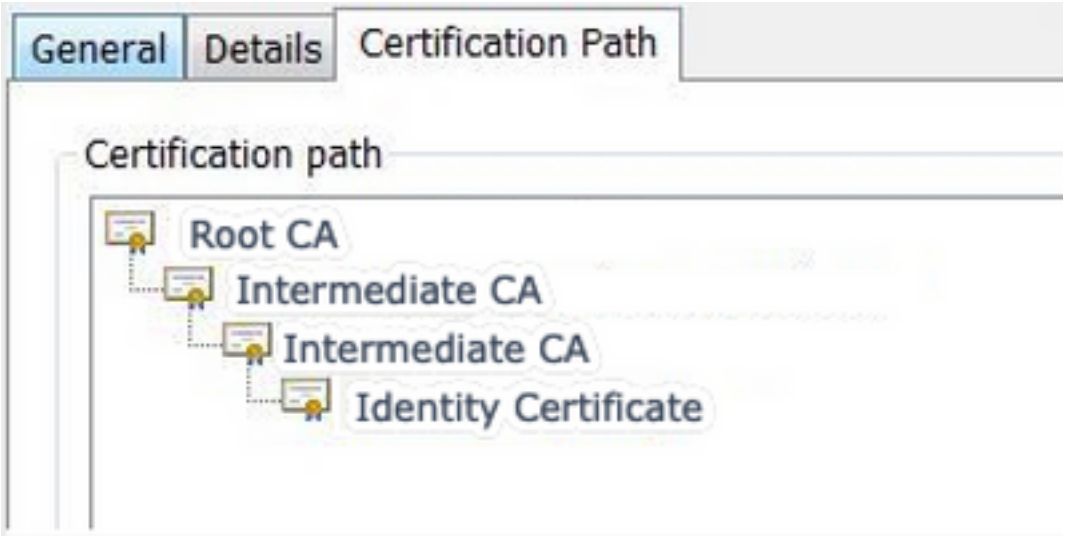
3단계. 이제 신뢰 지점이 있으므로 아래 명령을 사용하여 CSR 요청을 생성합니다.

```
 Crypto pki enroll CUBE_CA_CERT
```

화면의 질문에 답변한 다음 CSR 요청을 복사하여 파일에 저장한 다음 CA에 전송합니다.

4단계. 루트 인증서 체인에 중간 인증서가 있는지 확인해야 합니다.중간 인증 기관이 없는 경우 7단계로 건너뛰고, 그렇지 않은 경우 6단계로 진행합니다.

5단계. 루트 인증서를 보유할 신뢰 지점을 만들고, CUBE 인증서에 서명하는 CA가 있을 때까지 중간 CA를 보유할 신뢰 지점을 만듭니다(아래 이미지 참조).



이 예에서 1번째 레벨은 루트 CA이고, 2번째 레벨은 첫 번째 중간 CA이고, 3번째 레벨은 CUBE 인증서를 서명하는 CA입니다. 따라서 이러한 명령으로 첫 번째 2개의 인증서를 보유할 신뢰 지점을 생성해야 합니다.

```
Crypto pki trustpoint Root_CA_CERT
Enrollment terminal pem
Revocation-check none

Crypto pki authenticate Root_CA_CERT
Paste the X.64 based certificate here

Crypto pki trustpoint Intermediate_CA
Enrollment terminal
Revocation-check none

Crypto pki authenticate Intermediate_CA
```

6단계. CA 서명 인증서를 받은 후 신뢰 지점을 인증하려면 신뢰 지점이 CUBE 인증서 바로 앞에 CA 인증서를 보유해야 합니다.인증서를 가져올 수 있는 명령은

```
Crypto pki authenticate CUBE_CA_CERT
```

7단계. 인증서가 설치되면 CUBE 인증서를 가져오려면 이 명령을 실행해야 합니다

```
  Crypto pki import CUBE_CA_CERT cert
```

8단계. 생성한 신뢰 지점을 사용하도록 SIP-UA를 구성합니다.

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

9단계 다음과 같이 다이얼 피어를 구성합니다.

```
  dial-peer voice 9999 voip
  answer-address 35..
  destination-pattern 9999
  session protocol sipv2
```

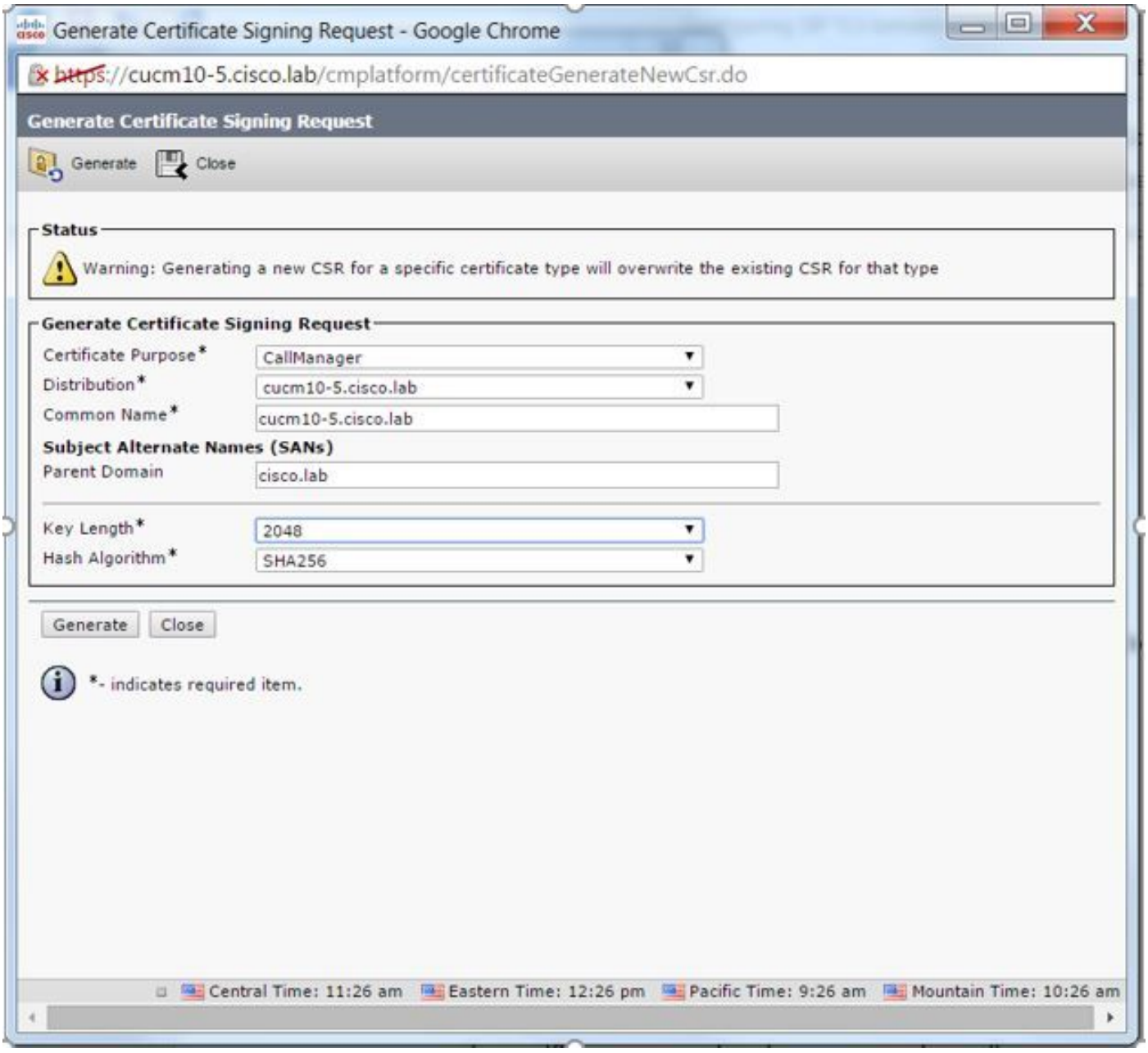
```
session target dns:cucm10-5
session transport tcp tls
voice-class sip options-keepalive
srtp
```

이렇게 하면 CUBE 구성이 완료됩니다.

10단계 이제 CUCM CSR을 생성하고 아래의 지침을 따릅니다.

- CUCM OS 관리자에게 로그인
- 보안 클릭
- 인증서 관리를 클릭합니다.
- Generate CSR(CSR 생성)을 클릭합니다.

CSR 요청은 다음과 같이 표시되어야 합니다.



11단계. CSR을 다운로드하고 CA에 전송합니다.

12단계. CA 서명 인증서 체인을 CUCM에 업로드합니다. 단계는 다음과 같습니다.

- 보안 을 클릭한 다음 인증서 관리를 클릭합니다.

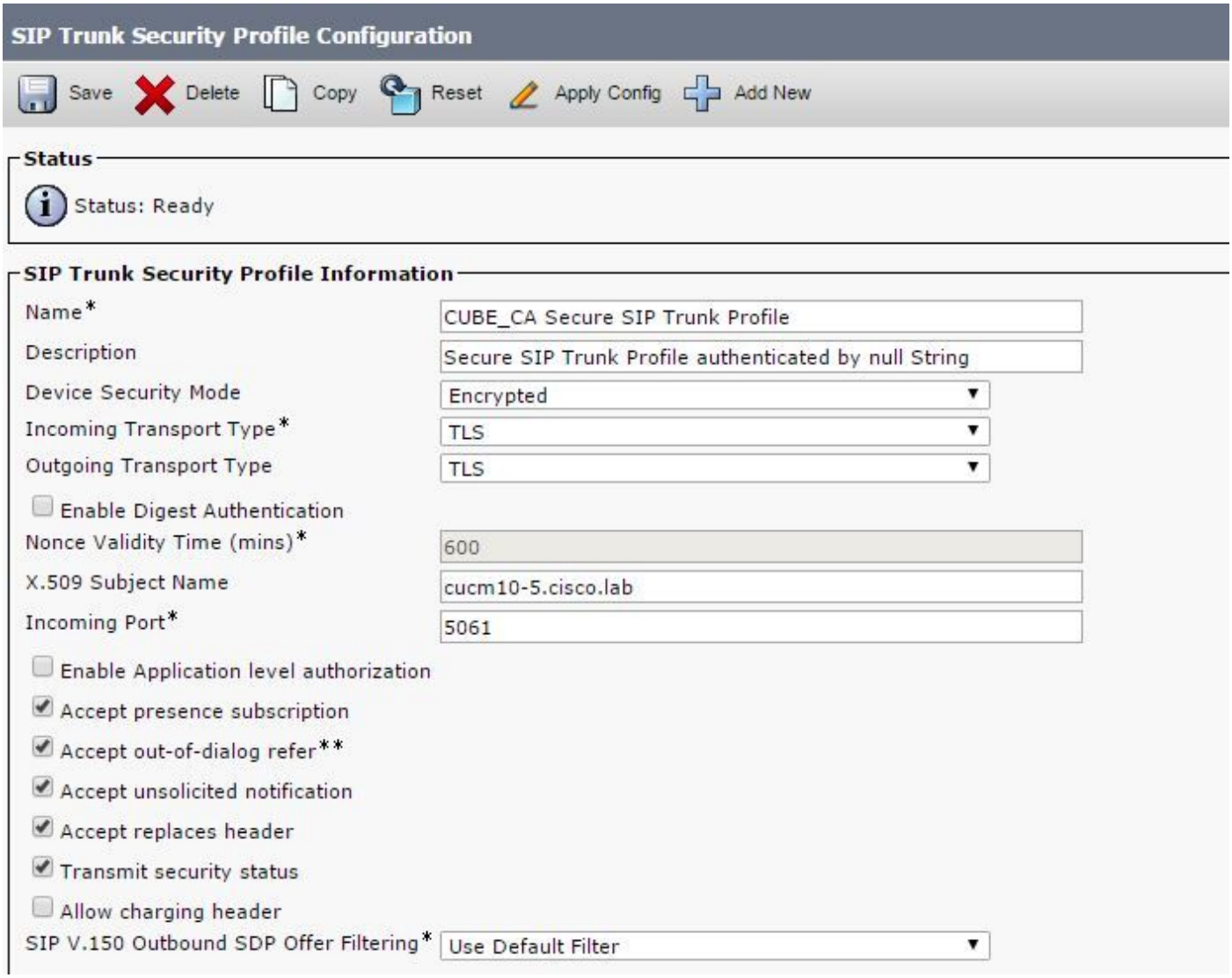
- 인증서/인증서 체인 업로드를 클릭합니다.
- 인증서 용도 드롭다운 메뉴에서 통화 관리자를 선택합니다.
- 파일을 찾습니다.
- 업로드를 클릭합니다.

13단계. CUCM CLI에 로그인하고 이 명령을 실행합니다.

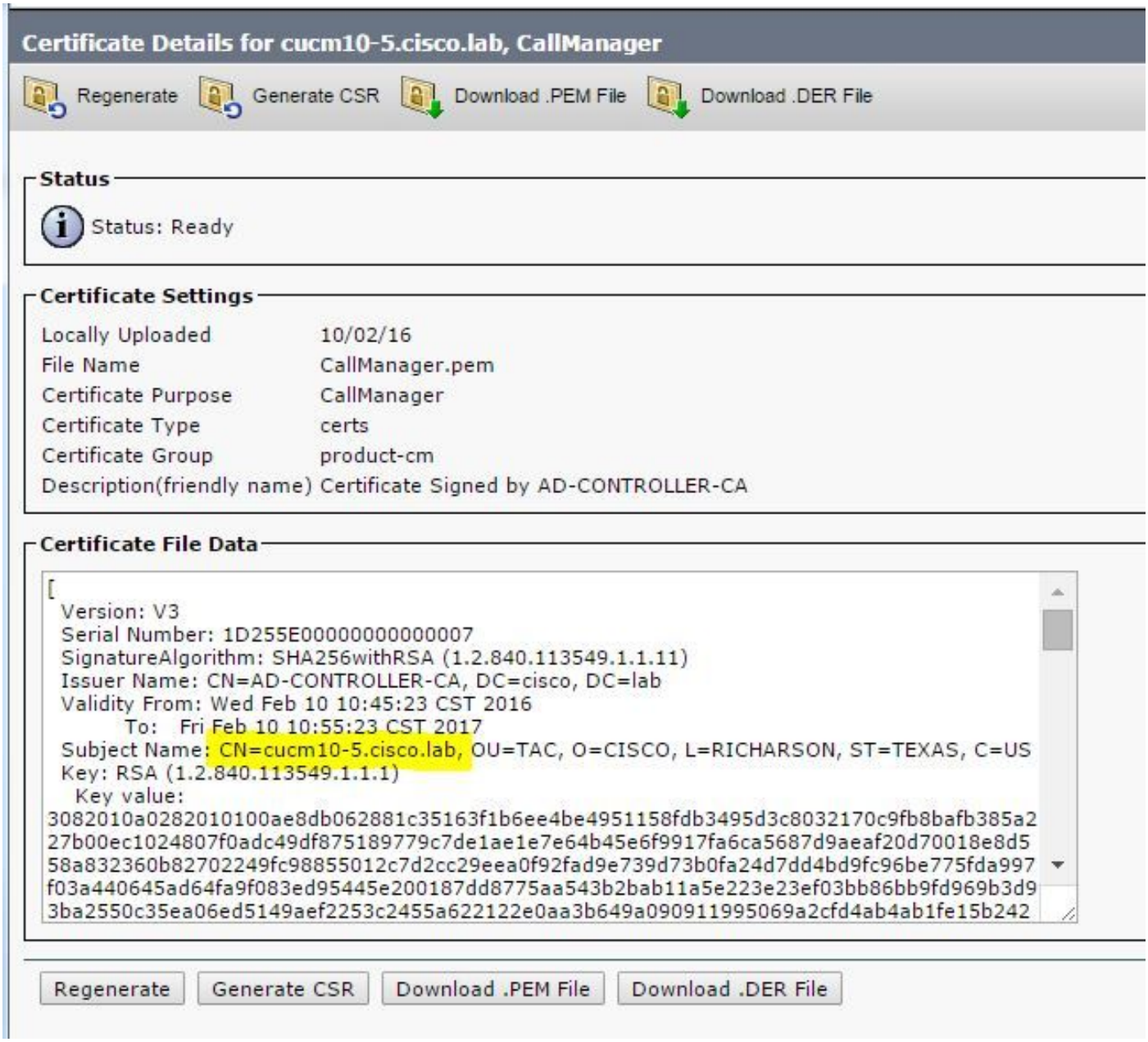
```
utils ctl update CTLFile
```

14단계. CUCM SIP 트렁크 보안 프로파일 구성

- 시스템, 보안, SIP 트렁크 보안 프로파일 을 차례로 클릭합니다.
- 이미지에 표시된 대로 프로파일을 구성합니다.



참고:이 경우 X.509 주체 이름은 이미지의 강조 표시된 부분에 표시된 CUCM 인증서 주체 이름과 일치해야 합니다.



15단계. CUCM에서 일반적으로 수행하는 것과 같이 SIP 트렁크를 구성합니다.

- SRTP Allowed(SRTP 허용) 확인란이 선택되어 있는지 확인합니다.
- 적절한 목적지 주소를 구성하고 포트 5060을 포트 5061로 교체해야 합니다.
- SIP 트렁크 보안 프로필에서 14단계에서 생성한 SIP 프로필 이름을 선택해야 합니다.

다음을 확인합니다.

현재 모든 컨피그레이션이 정상이면

CUCM에서 SIP 트렁크 상태는 이미지에 표시된 대로 Full Service를 표시합니다.

Name	Description	Calling Search Space	Device Pool	Route Pattern	Partition	Route Group	Priority	Trunk Type	SIP Trunk Status	SIP Trunk Duration
ISR4451-B			G711-Secure					SIP Trunk	Full Service	Time In Full Service: 0 day 0 hour 0 minute

CUBE에서 다이얼 피어는 다음 상태를 표시합니다.

```
TAG    TYPE  MIN  OPER PREFIX    DEST-PATTERN      FER THRU SESS-TARGET    STAT PORT
KEEPALIVE

9999   voip  up   up             9999               0  syst dns:cucm10-5                active
```

이 동일한 프로세스가 다른 라우터에 적용되며 유일한 차이점은 CUCM 인증서를 업로드하는 단계 대신 서드파티에서 제공한 인증서를 업로드하는 것입니다.

문제 해결

CUBE에서 이러한 디버깅 사용

```
debug crypto pki api
debug crypto pki callbacks
debug crypto pki messages
debug crypto pki transactions
debug ssl openssl errors
debug ssl openssl msg
debug ssl openssl states
debug ip tcp transactions
```