모바일 및 원격 액세스 인증서 요구 사항 및 ATS 기록 이해

목차

소개

이 문서에서는 모바일 및 원격 액세스를 위한 CUCM의 인증서 업로드 요구 사항에 대해 설명합니다.

배경 정보

Cisco Expressway는 ATS(Apache Traffic Server)를 사용합니다. 트래픽 서버는 접근 솔루션에서 매우 중요한 구성 요소이며 주로 다음 기능에 사용됩니다.

- 인증서 확인: MRA 서비스용 CUCM(Cisco Unified Communications Manager), IM & Presence 및 Unity 서버 노드의 인증서 확인을 수행합니다.
- 프록시 및 캐싱: HTTP/HTTPS 트래픽에 대한 빠르고 확장 가능한 캐싱 프록시 서버 역할을 합니다.

Expressway 버전 14.0.2

트래픽 서버(ATS)는 MRA 프로비저닝 중 CUCM과 통신할 때 '인증서 확인'을 약간 적용하기 시작합니다.

요구 사항은 CSCvz45074에 문서화되었으며 여기서 Expressway Core 서버 인증서를 서명한 루트 인증서는 Tomcat-Trust 및 Callmanager Trust로서 CUCM에 업로드해야 합니다. https://cdetsng.cisco.com/summary/#/defect/CSCvz45074입니다.

- 트래픽 서버는 인증서 확인을 적용합니다.
- X14.0.2 릴리스로 업그레이드하기 전에 이 인증서 요구 사항이 충족되었는지 확인하십시오.

요구 사항 - UCM(Unified Communications Manager)이 비보안 모드에 있는 경우에도 Expressway-C 인증서에 서명한 CA(Certificate Authority) 체인(루트 + 중간)을 CUCM의 tomcat-trust 및 CallManager-trust 목록에 추가해야 합니다.

Reason(이유) - Expressway의 트래픽 서버 서비스는 서버 UCM에서 요청할 때마다 인증서를 전송합니다. 이러한 요청은 8443 이외의 포트(예: 포트 6971, 6972 등)에서 실행되는 서비스에 대한 것입니다. 이는 UCM이 비보안 모드에 있는 경우에도 인증서 확인을 적용합니다. 자세한 내용은 Expressway 구축 가이드를 통한 모바일 및 원격 액세스를 참조하십시오.

14.0.8 이전 버전의 동작

Expressway-C와 Unified Communication 노드 간의 보안 HTTPS 양방향 연결을 처리하는 Expressway-C의 트래픽 서버에서 원격 끝점이 제공한 인증서를 확인하지 못했습니다. MRA 컨피그레이션에서는 Configuration(컨피그레이션) > Unified Communications > Unified CM servers/IM and Presence Service nodes/Unity Connection servers(Unified CM 서버/IM 및 프레즌스 서비스 노드/Unity 연결 서버)에서 CUCM, IM&P 또는 Unity 서버를 추가할 때 TLS Verify Mode(TLS 확인 모드)의 컨피그레이션을 통해 TLS 인증서를 확인하도록 하는 옵션이 있습니다. 구성 옵션은 다음 스크린샷에 표시되어 SAN의 FQDN 또는 IP는 물론 인증서의 유효성 및 신뢰받는 CA에서 서명되었는지 여부를 확인합니다.

Expressway 트러스트 스토어에서 CN 이름이 같은 인증서 2개를 로드할 수 없는 알려진 문제도 있었습니다. 이러한 제한으로 인해 두 가지 문제가 발생했습니다.

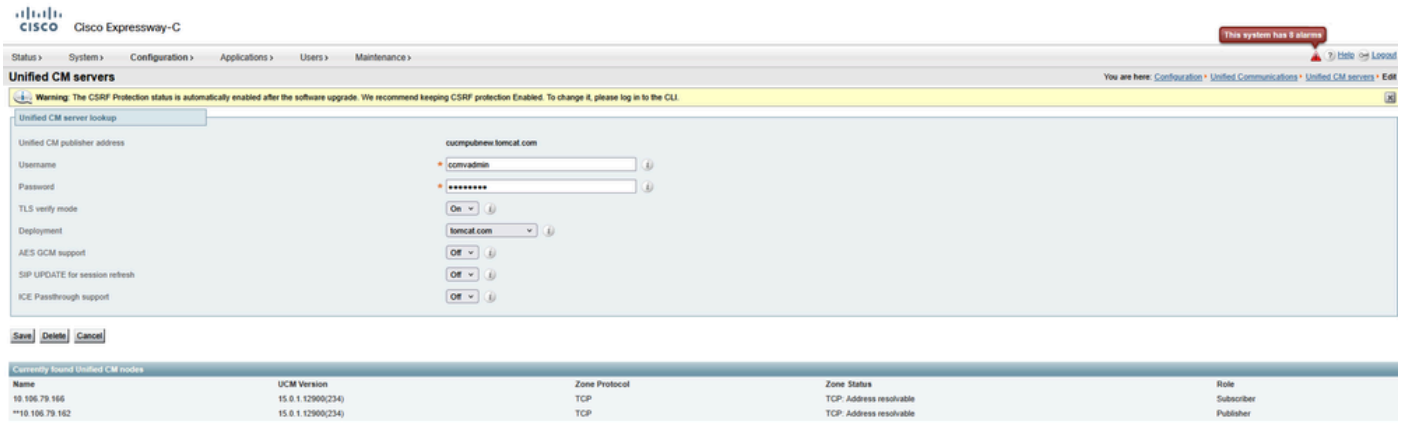
1. Expressway Trust 저장소에서 통화 관리자 인증서를 로드하도록 선택한 경우 CUCM을 추가하는 동안 TLS 확인 'On'이 실패합니다.

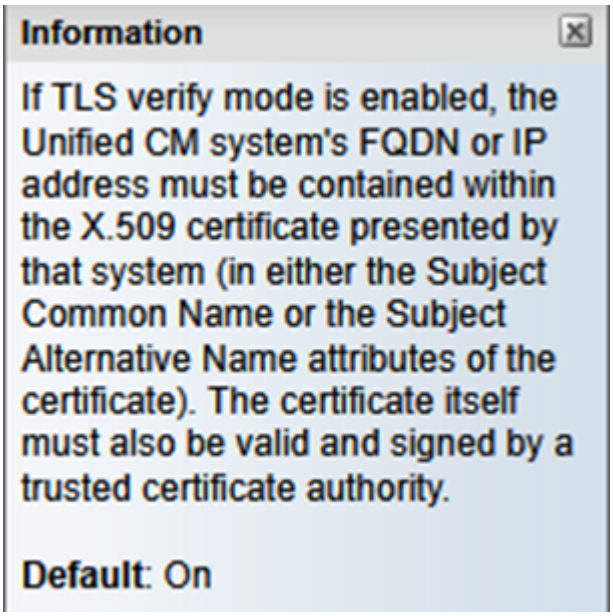
2: Expressway Trust 저장소에서 Tomcat 인증서를 로드하도록 선택한 경우 5061의 보안 sip 등록이 실패합니다.

이 동작은 CSCwa12894에 설명되어 있습니다.

또한 이 TLS 인증서 확인 검사는 CUCM/IM&P/Unity 서버 검색에서만 수행되며 MRA 클라이언트 프로비저닝 중에는 수행되지 않습니다.

이 구성의 단점은 추가한 게시자 주소에 대해서만 확인한다는 것입니다. 게시자 노드의 데이터베이스에서 가입자 노드 정보(FQDN 또는 IP)를 검색하므로 가입자 노드의 인증서가 올바르게 설정되었는지 검증하지 않습니다.





버전 14.0.8 이상에서 동작

X14.0.8 버전 이후부터 Expressway 서버는 트래픽 서버를 통해 생성되는 모든 단일 HTTPS 요청에 대해 TLS 인증서 확인을 수행합니다. 이는 CUCM/IM&P/Unity 노드를 검색하는 동안 TLS Verify Mode(TLS 확인 모드)가 'Off(해제)'로 설정된 경우에도 이를 수행함을 의미합니다. 검증에 성공하지 못하면 TLS 핸드셰이크가 완료되지 않고 요청이 실패하기 때문에 리던던시, 장애 조치 문제 또는 완전한 로그인 실패와 같은 기능이 손실될 수 있습니다. 또한 TLS Verify Mode(TLS 확인 모드)가 'On(켜기)'으로 설정된 경우, 모든 연결이 나중에 예에서 설명한 대로 제대로 작동한다는 보장은 없습니다.

Expressway에서 CUCM/IM&P/Unity 노드에 대해 확인하는 정확한 인증서는 MRA 가이드의 섹션에 나와 있습니다.

https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/expressway/config_guide/X15-0/mra/exwy_b_mra-deployment-guide-x150.pdf

섹션

Certificate Requirements(인증서 요건) > Certificate Exchange Requirements(인증서 교환 요건)

Expressway-Core와 CUCM 간의 통신 방식이 다음과 같이 변경되므로 다음 사항이 보장되어야 합니다.

1. 모바일 및 원격 액세스에 CA 서명 인증서를 사용하는 것이 좋습니다.

2. 각 Unified CM 클러스터는 Expressway-C 인증서를 신뢰해야 합니다. 각 클러스터에 대해 다음을 확인합니다.

- 혼합 모드가 활성화된 경우 — Expressway-C 인증서를 Unified CM의 CallManager-trust 및 Tomcat-trust 저장소에 설치해야 합니다.
- 혼합 모드가 비활성화된 경우 — Expressway-C 인증서를 서명하는 루트 CA 인증서를 Unified CM의 CallManager-trust 및 Tomcat-trust 저장소에 설치해야 합니다. 그런 다음 다음을 다시

시작합니다. · Tomcat 서비스 · CallManager 서비스 · HA 프록시 서비스(Tomcat에서 TLS를 사용하는 경우)

Expressway - Core에서 다음 작업이 수행되는지 확인합니다.

- Expressway-C는 각 Unified CM 및 IM and Presence Service 클러스터에서 제공하는 인증서를 신뢰해야 합니다.

Expressway-C의 신뢰 저장소에는 모든 UC 클러스터에 대한 Unified CM 및 IM and Presence Service 인증서에 서명하는 루트 CA 인증서가 포함되어야 합니다.

참고: UCM이 비보안 모드에서 작동 중인 경우에도 Expressway-C 인증서를 서명하는 데 사용되는 모든 루트 및 중간 CA 인증서 또는 전체 CA 체인을 Cisco UCM(Unified Communications Manager)의 Tomcat-trust 및 CallManager-trust 목록에 추가해야 합니다.

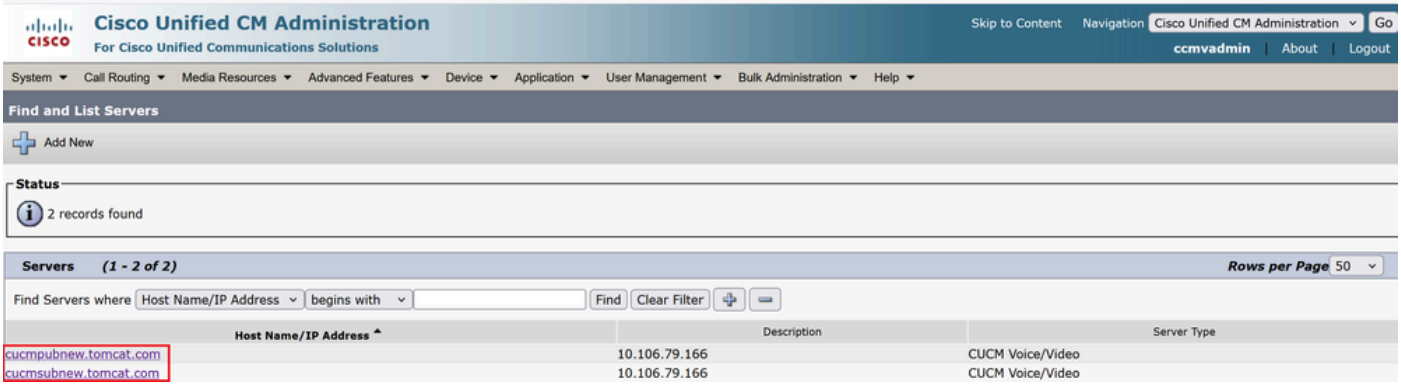
Reason(이유) - Expressway의 트래픽 서버 서비스는 서버(UCM)가 요청할 때마다 인증서를 전송합니다. 이러한 요청은 8443 이외의 포트(예: 포트 6971, 6972 등)에서 실행되는 서비스에 대한 것입니다. 이는 UCM이 비보안 모드에 있는 경우에도 인증서 확인을 적용합니다.

System(시스템) > Server(서버)에서 CUCM 주소를 추가하는 방법은 Configuration(컨피그레이션) > Unified Communications(Unified Communications) > Unified CM servers/IM and Presence Service(Unified CM 서버/IM and Presence 서비스) 노드 아래에서 Expressway Core에 CUCM/IMP를 추가하는 데 매우 중요한 역할을 합니다.

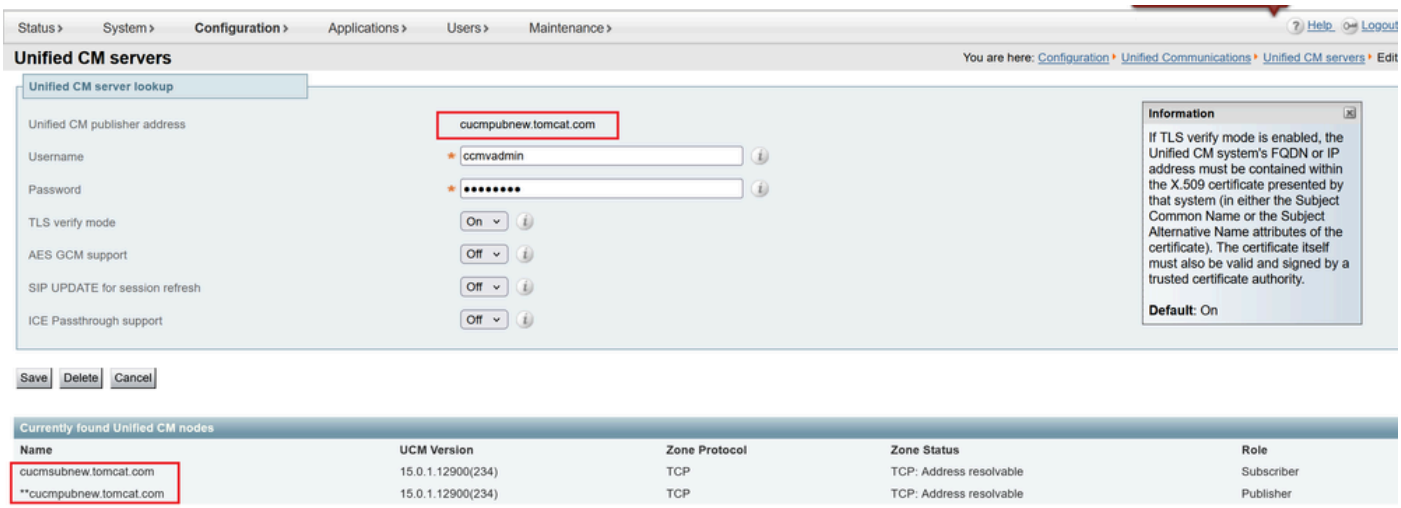
CUCM은 항상 호스트 이름 또는 IP 주소가 아니라 FQDN과 함께 추가해야 합니다. CUCM이 표시된 경우 System(시스템) > Server(서버)에 Hostname/IP address(호스트 이름/IP 주소)로 추가됩니다.

tls 핸드셰이크 중에 TLS 확인 'On'이 실패하고 Expressway-Core에 CUCM 클러스터가 추가되지 않습니다.

이 그림에서는 호스트 이름으로 추가된 CUCM을 보여줍니다.



이 그림에서는 FQDN이 있는 Expressway-Core에 추가된 CUCM(TLS 확인 모드 = ON)을 보여 줍니다.

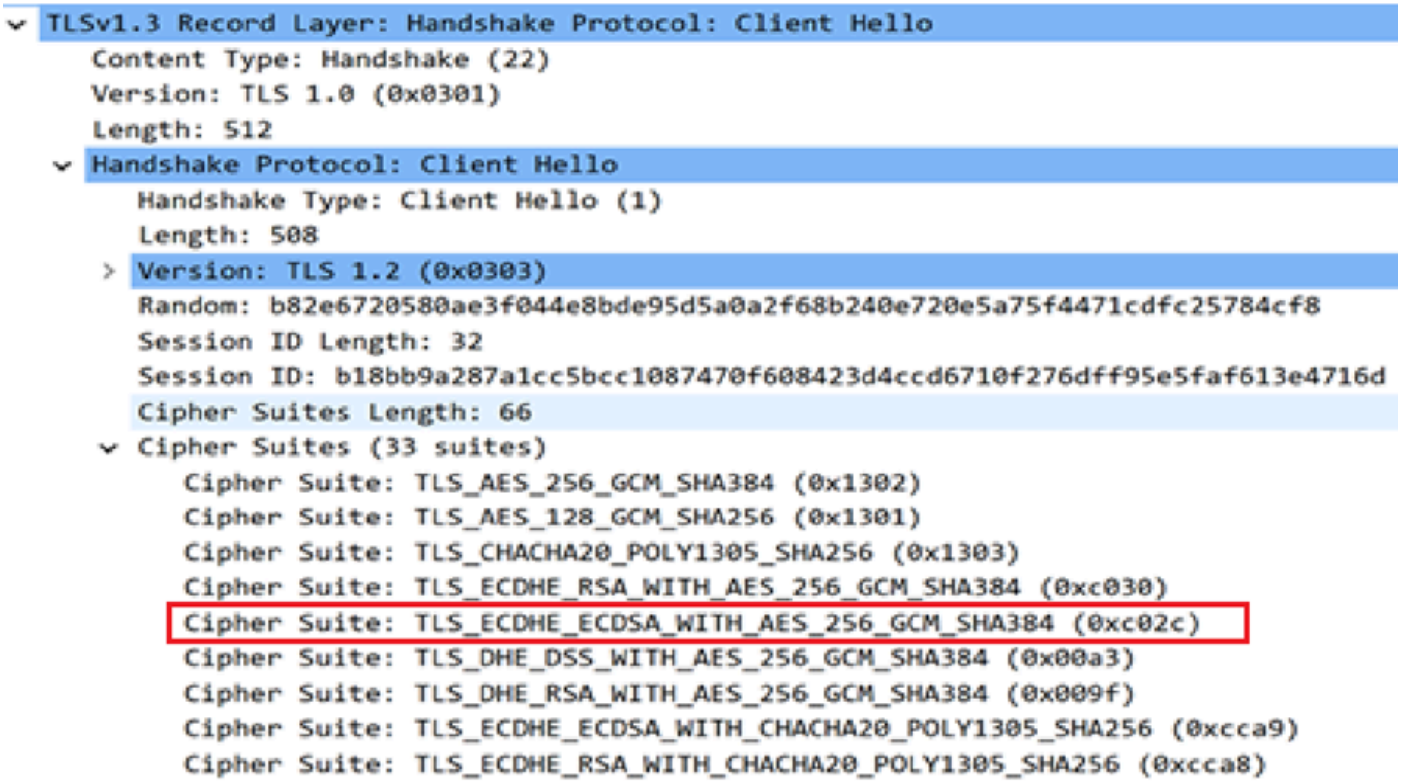


또한 X14.2에 변경 사항이 도입되어 TLS 핸드셰이크(클라이언트 hello) 중에 서로 다른 기본 설정 순서로 암호를 제공합니다. 이는 업그레이드 경로에 따라 달라졌으며 소프트웨어 업그레이드 후 예기치 않은 TLS 연결이 발생했습니다. 이는 TLS 핸드셰이크 중 업그레이드 전에 CUCM에서 Cisco Tomcat 또는 Cisco CallManager 인증서를 요청했기 때문일 수 있습니다. 그러나 업그레이드 후 ECDSA 변형(RSA보다 더 안전한 암호 변형)을 요청했습니다. Cisco Tomcat-ECDSA 또는 Cisco CallManager-ECDSA 인증서는 다른 CA에서 서명하거나 자체 서명 인증서(기본값)만 사용할 수 있습니다.

이 암호 기본 설정 순서 변경은 Expressway X14.2.1 릴리스 노트에 나와 있는 업그레이드 경로에 따라 달라지므로 항상 귀하와 관련이 있는 것은 아닙니다. 요약하면, ECDHE-RSA-AES256-GCM-SHA384 앞에 추가되는지 여부에 관계없이 각 암호 목록에 대한 Maintenance(유지 관리) > Security(보안) > Ciphers(암호)에서 볼 수 있습니다. 그렇지 않은 경우 RSA 암호보다 최신 ECDSA 암호를 선호합니다. 그러면 RSA에서 더 높은 우선 순위를 가진 이전 RSA와 같은 동작이 적용됩니다.

다음 스크린샷은 Client hello의 TLS 협상 메시지 중에 Expressway 코어가 광고한 빨간색 상자 ECDSA 암호, #IF TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384가 서버 hello의 CUCM(Remote responder)에 의해 선택되면 TLS 협상이 실패하는 것을 보여줍니다.

Responder의 ROOT CA 인증서 또는 실제 ECDSA 인증서, 즉 CUCM이 Expressway Trust 저장소에 설치되어 있지 않습니다.



또는 ECDSA가 우선하지 않도록 Expressway 암호를 수정할 수도 있습니다.

1. GCM-Sha384 open SSL 문자열을 추가하여 SIP 암호를 수정합니다.

"ECDHE-RSA-AES256-GCM-SHA384:ECDH:EDH:HIGH:.....:!MD5:!PSK:!eNULL:!aNULL:!aDH"

2. 마지막 기본 설정에서 암호를 이동하려면 +를 추가하고 ECDSA를 영구적으로 사용하지 않도록 설정하려면 !을(를) 추가합니다.

암호: "ECDH:EDH:HIGH:-
AES256+SHA:!MEDIUM:!LOW:!3DES:!MD5:!PSK:!eNULL:!aNULL:!aDH:+ECDSA"

3. CUCM에서 ECDSA 인증서를 서명한 루트 및 중간 CA 인증서를 추가하거나 Expressway 트러스트 스토어에서 Tomcat-ECDSA 인증서를 추가합니다(경우에 따라).

그러나 암호화 우선 순위의 변화, 업그레이드 후, MRA 구축이 중단될 수 있으므로, TAC는 앞서 언급한 해결 방법을 수행하여 작업을 다시 수행해야 합니다.

TLS 1.3이 도입되면서 wireshark에서 어떤 인증서가 교환되는지 확인하기가 더욱 어려워졌다.

버전 x15.3의 동작

SIP 인터페이스에만 RSA 또는 ECDSA 암호를 사용하도록 선택할 수 있습니다.

X15.x TLS 1.3이 적용되었습니다. 필드에서 볼 수 있듯이, RSA 알고리즘은 대부분 ECDSA를 통해 선택됩니다. x15.2로 업그레이드하는 고객은

이 명령 집합을 사용하는 RSA와 ECDSA 알고리즘 사이:

xConfiguration SIP 고급 TlsSignatureAlgoPrefRsa: 켜기/끄기

TlssignatureAlgoPrefRSA는 SIP 인터페이스에 TLS 1.3이 있는 경우에만 작동합니다.

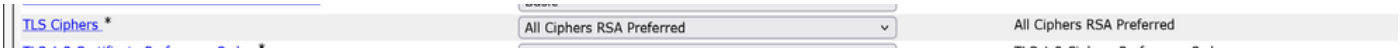
xConfiguration SIP 고급 SipTls버전: "TLSv1.3"

참고: 현재 SIP 인터페이스에 대해서만 사용할 수 있습니다. 8443의 Traffic Server 및 Tomcat 고려 사항은 앞에서 설명한 대로 변경되지 않습니다.

Expressway에서 CUCM으로 '클라이언트 hello' 중에 전송되는 암호 정장은 RSA를 선택할 때 표시됩니다.

- 서명 알고리즘: rsa_pss_rsae_sha512(0x0806)
- 서명 알고리즘: rsa_pss_rsae_sha384(0x0805)
- 서명 알고리즘: rsa_pss_rsae_sha256(0x0804)
- 서명 알고리즘: ecdsa_secp521r1_sha512(0x0603)
- 서명 알고리즘: ecdsa_secp384r1_sha384(0x0503)
- 서명 알고리즘: ecdsa_secp256r1_sha256(0x0403)

이전 컨피그레이션은 CUCM to TLS 암호에서 Enterprise Parameters(엔터프라이즈 매개변수) > Security Parameters(보안 매개변수) 아래에서 선택한 컨피그레이션에 대해 함께 작동합니다.



또한 Expressway-C와 CUCM 간의 TLS 1.3을 통한 깨진 TLS 핸드셰이크 중에는 진단 로그 또는 PCAP에 인쇄된 오류가 큰 도움이 되지 않는다는 점에 유의해야 합니다. TAC에서 작업하는 동안 이러한 디버그를 활성화할 가치가 있으므로 구성 요소에서 문제를 해결하기 위해 명확한 오류를 인쇄합니다.

xConfiguration Logger Developer.trafficserver.http 수준: "디버그"
xConfiguration Logger Developer.trafficserver.http_trans 레벨: "디버그"
xConfiguration Logger Developer.trafficserver.iocore 수준: "디버그"
xConfiguration Logger Developer.trafficserver.ssl 레벨: "디버그"

Callmanager가 하나의 인증서를 여러 서비스와 공유할 경우 예상되는 사항

CUCM에서 인증서를 재사용하면 상황이 약간 변경됩니다.

CUCM 14.0부터는 Tomcat 및 Tomcat ECDSA 인증서를 Call Manager 및 Call Manager ECDSA로 재사용할 수 있습니다.

Tomcat 인증서는 Callmanager 인증서로 재사용할 수 있습니다.

Tomcat-ECDSA 인증서는 Callmanager-ECDSA 인증서로 재사용할 수 있습니다.

이것은 인생을 편하게 해줍니다.

1. 이제 CUCM의 여러 서비스에서 하나의 인증서를 사용하므로 인증서 비용이 절감됩니다.

2. 인증서의 관리가 부실할 것

3. Expressway-Core 트러스트 스토어에 Tomcat/Callmanager 또는 Tomcat-ECDSA/Callmanager-ECDSA 인증서를 업로드해야 하는 경우(어떤 이유로든) 한 개의 인증서만 업로드하면 됩니다. 동일한 CN 이름 문제가 발생하는 문제는 없습니다(이 문서의 앞부분에서 설명).

참고: 인증서의 재사용은 Tomcat 및 Tomcat-ECDSA가 다중 SAN 인증서인 경우에만 발생합니다.

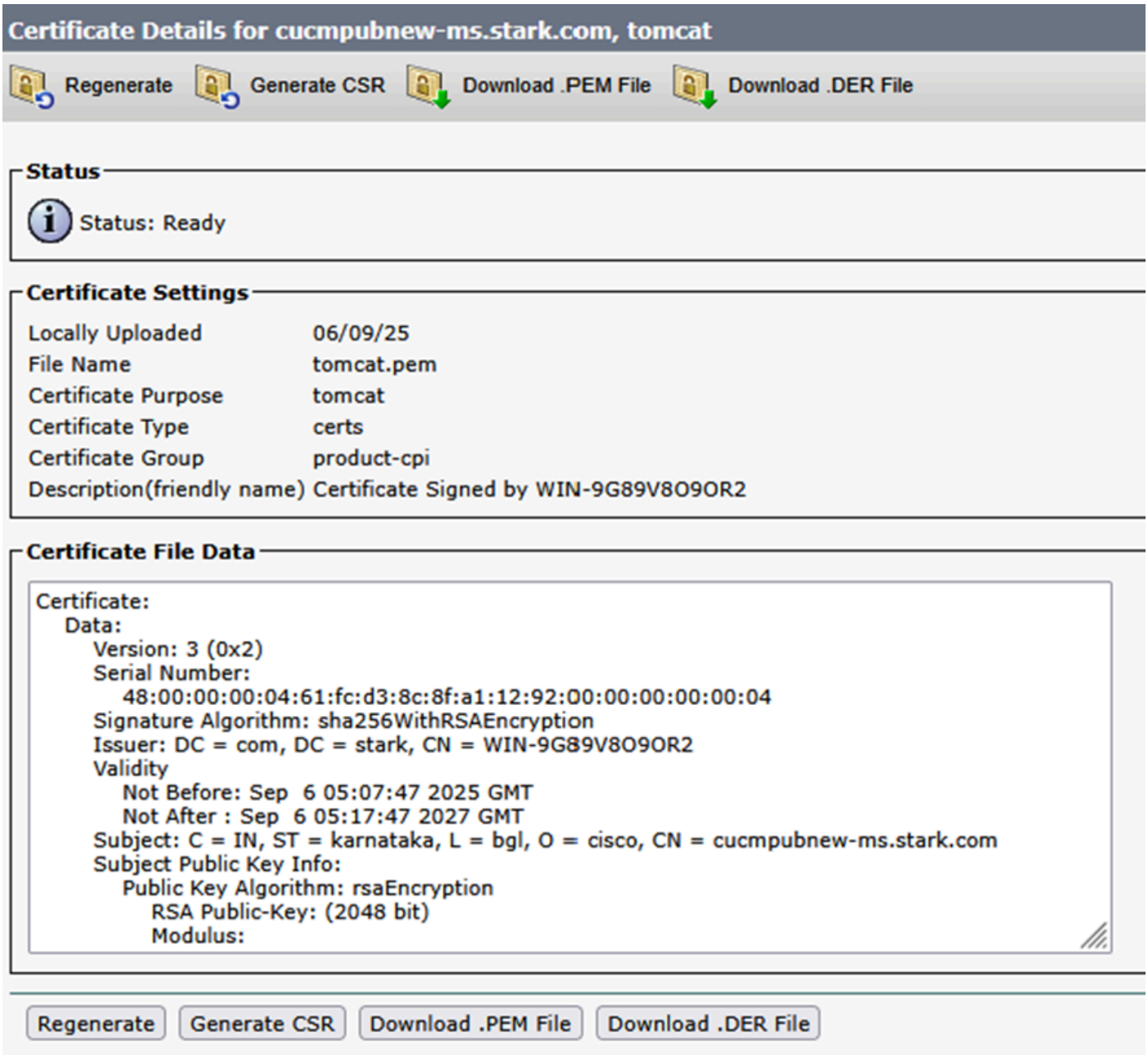
Post Reuse(다시 사용 후), Callmanager 및 Callmanager ECDSA 서버 인증서는 CUCM 트러스트 저장소에 표시되지 않습니다. 다음 명령을 실행하여 CLI에서 인증서 재사용을 검증할 수 있습니다.

show cert own CallManager

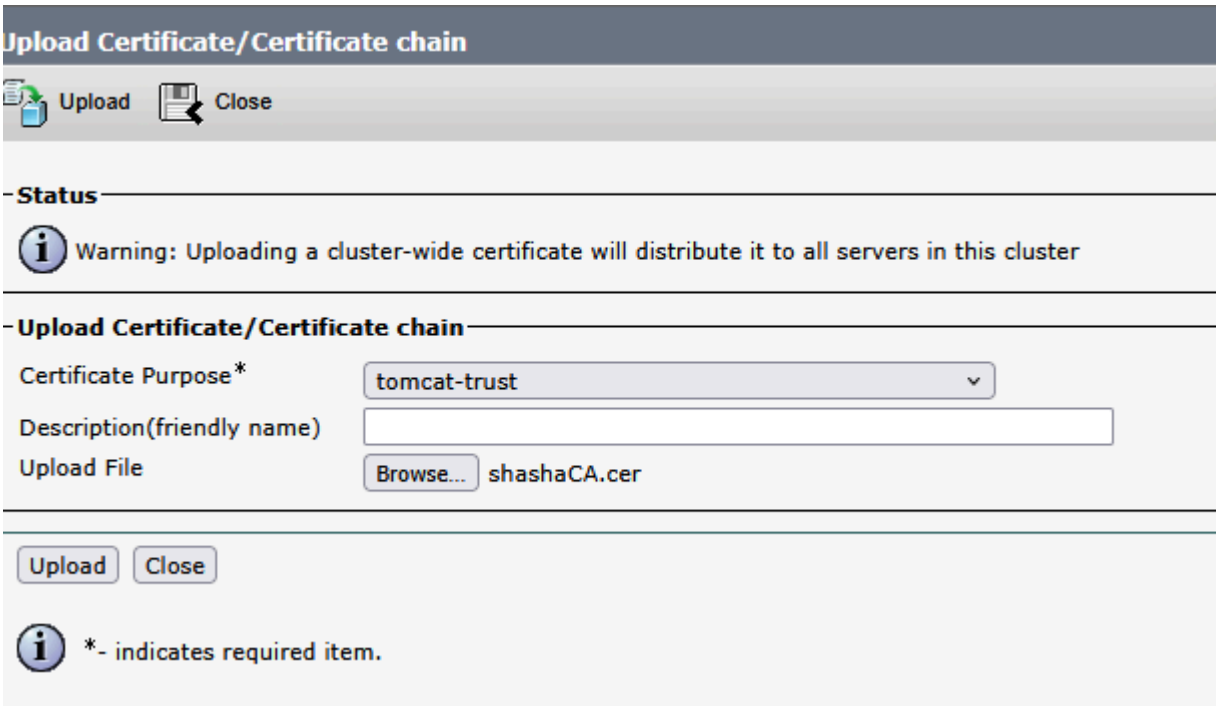
show cert own tomcat

인증서 재사용 단계

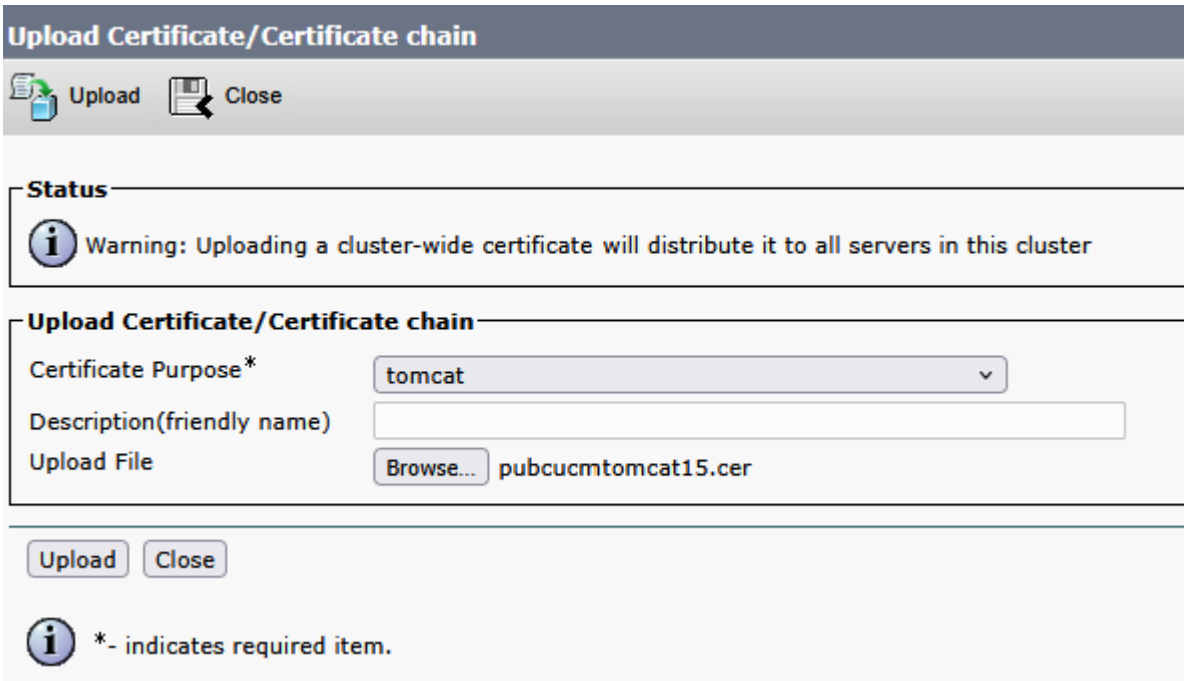
Tomcat CSR pub 추가 생성



CUCM에서 Tomcat 인증서를 Tomcat-trust로 서명할 CA 인증서를 업로드합니다.



Tomcat 인증서가 서명되면 게시자에 업로드합니다. 프롬프트에 따라 관련 서비스를 다시 시작합니다.



Tomcat 인증서가 서명되면 게시자에 업로드합니다. 프롬프트에 따라 관련 서비스를 다시 시작합니다.

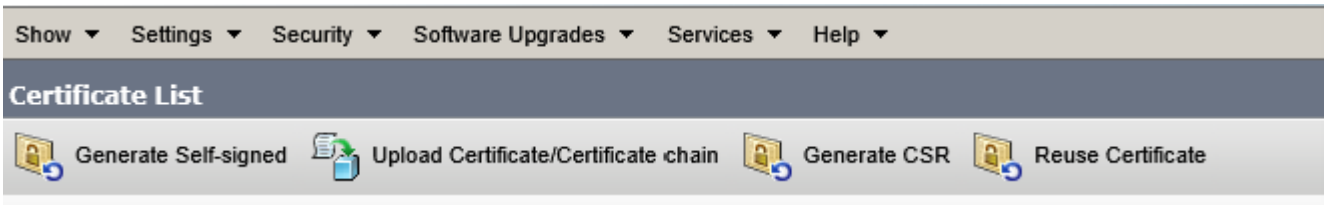
Success: 인증서가 업로드되었습니다. 최신 백업에 업로드된 인증서가 포함되도록 재해 복구 백업을 수행합니다.
모든 클러스터 노드(UCM/IMP)에서 CLI 'utils service restart Cisco Tomcat'을 사용하여 Cisco Tomcat 웹 서비스를 재시작합니다. 모든 UCM 클러스터 노드에서 CLI 'utils service restart Cisco

UDS Tomcat and utils service restart Cisco AXL Tomcat'을 사용하여 Cisco UDS Tomcat 및 Cisco AXL Tomcat 웹 서비스를 재시작합니다. 또한 게시자 노드에서 Cisco DRF 마스터 및 Cisco DRF 로컬 서비스를 다시 시작합니다. 가입자 노드에서 Cisco DRF 로컬 서비스만 다시 시작합니다.

이제 Tomcat 인증서가 CA에 의해 서명되었습니다.

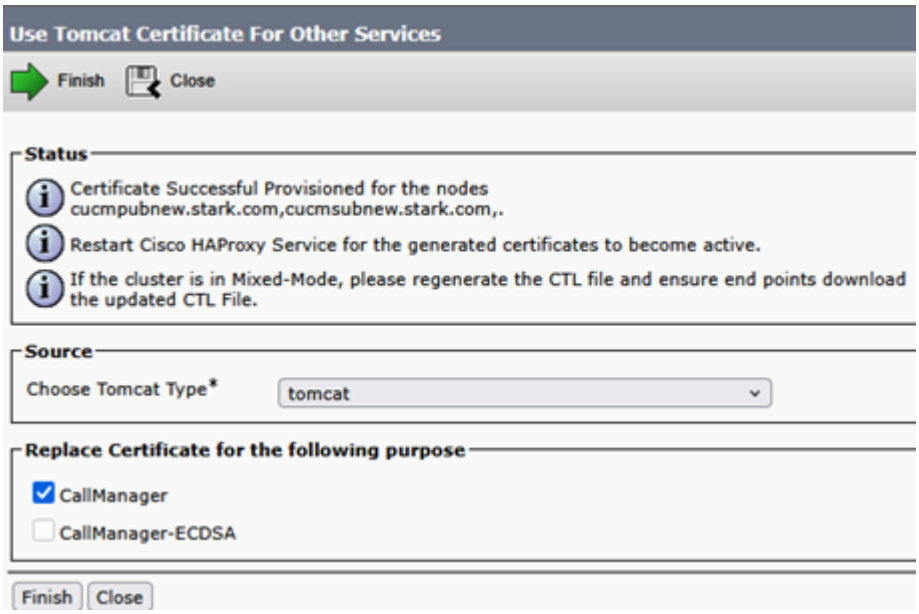
Tomcat 인증서를 이제 Callmanager 인증서로 재사용하려면

Reuse Certificate(인증서 재사용)를 클릭합니다.



드롭다운에서 Tomcat을 선택하고 Callmanager 인증서를 선택합니다.

Finish(마침)를 클릭합니다.



Tomcat 인증서가 Callmanager 인증서로 다시 사용됩니다. 이는 CLI에서 검증할 수 있습니다.

Callmanager 인증서 일련 번호(SN): 56:ff:6c:71:00:00:00:00:00:0d

```
admin:show cert own CallManager
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            56:ff:6c:71:00:00:00:00:00:0d
        Signature Algorithm: sha384WithRSAEncryption
        Issuer: DC = com, DC = RICKY200, CN = RICKY200-TMS-CA
        Validity
            Not Before: Oct 24 08:44:34 2025 GMT
            Not After : Oct 24 08:54:34 2027 GMT
        Subject: C = IN, ST = karnataka, L = bgl, O = cisco, CN = cucmpubnew-ms.
tomcat.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (2048 bit)
                Modulus:
                    00:b4:a6:fa:8f:9a:c3:32:02:74:fa:e9:92:30:de:
                    6e:3b:70:cd:d7:4e:64:e4:71:04:fe:17:80:0d:5b:
                    44:d1:7f:00:63:69:4a:5c:1a:1b:75:0c:1a:d6:ce:
                    10:3f:01:e2:d0:f1:75:33:57:b7:0a:71:e1:60:d1:
                    89:3c:e8:a4:8c:3e:30:69:4d:4e:98:da:b8:5d:dd:
                    23:8c:4d:69:90:69:9d:43:74:84:20:a8:9f:45:dc:
                    5a:aa:7b:c8:d1:d0:6f:05:13:d8:99:58:0e:49:7b:
Press <enter> for 1 line, <space> for one page, or <q> to quit
```

Tomcat 인증서 SN: 56:ff:6c:71:00:00:00:00:00:0d

```
admin:show cert own tomcat
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            56:ff:6c:71:00:00:00:00:00:0d
        Signature Algorithm: sha384WithRSAEncryption
        Issuer: DC = com, DC = RICKY200, CN = RICKY200-TMS-CA
        Validity
            Not Before: Oct 24 08:44:34 2025 GMT
            Not After : Oct 24 08:54:34 2027 GMT
        Subject: C = IN, ST = karnataka, L = bgl, O = cisco, CN = cucmpubnew-ms.tomcat.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (2048 bit)
                Modulus:
                    00:b4:a6:fa:8f:9a:c3:32:02:74:fa:e9:92:30:de:
                    6e:3b:70:cd:d7:4e:64:e4:71:04:fe:17:80:0d:5b:
                    44:d1:7f:00:63:69:4a:5c:1a:1b:75:0c:1a:d6:ce:
                    10:3f:01:e2:d0:f1:75:33:57:b7:0a:71:e1:60:d1:
                    89:3c:e8:a4:8c:3e:30:69:4d:4e:98:da:b8:5d:dd:
                    23:8c:4d:69:90:69:9d:43:74:84:20:a8:9f:45:dc:
                    5a:aa:7b:c8:d1:d0:6f:05:13:d8:99:58:0e:49:7b:
Press <enter> for 1 line, <space> for one page, or <q> to quit
```

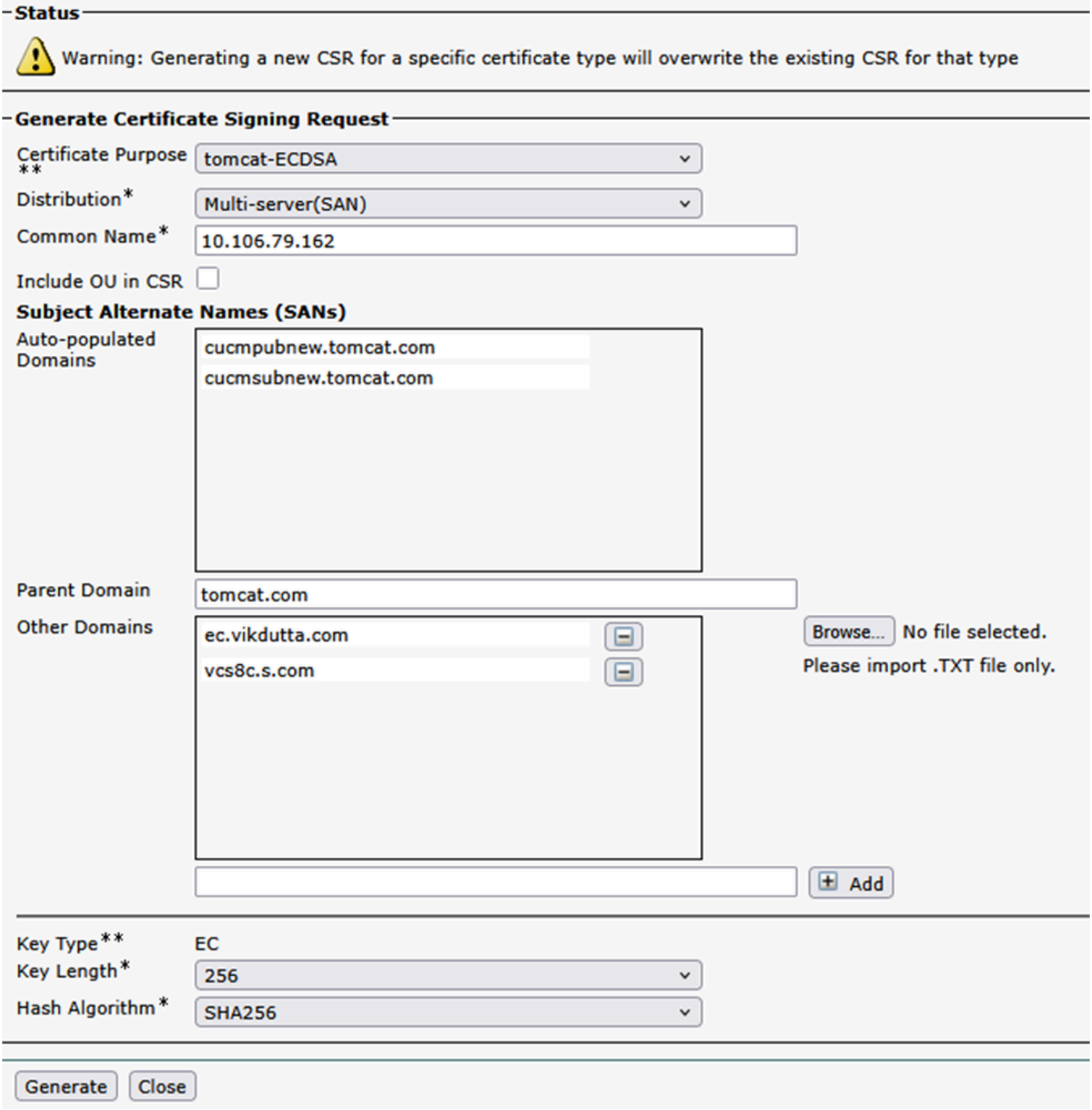
가입자에 대해 동일한 단계를 수행합니다.

이제 ECDSA 인증서를 서명하여 CallManager-ECDSA로 다시 사용할 수 있도록 하겠습니다.

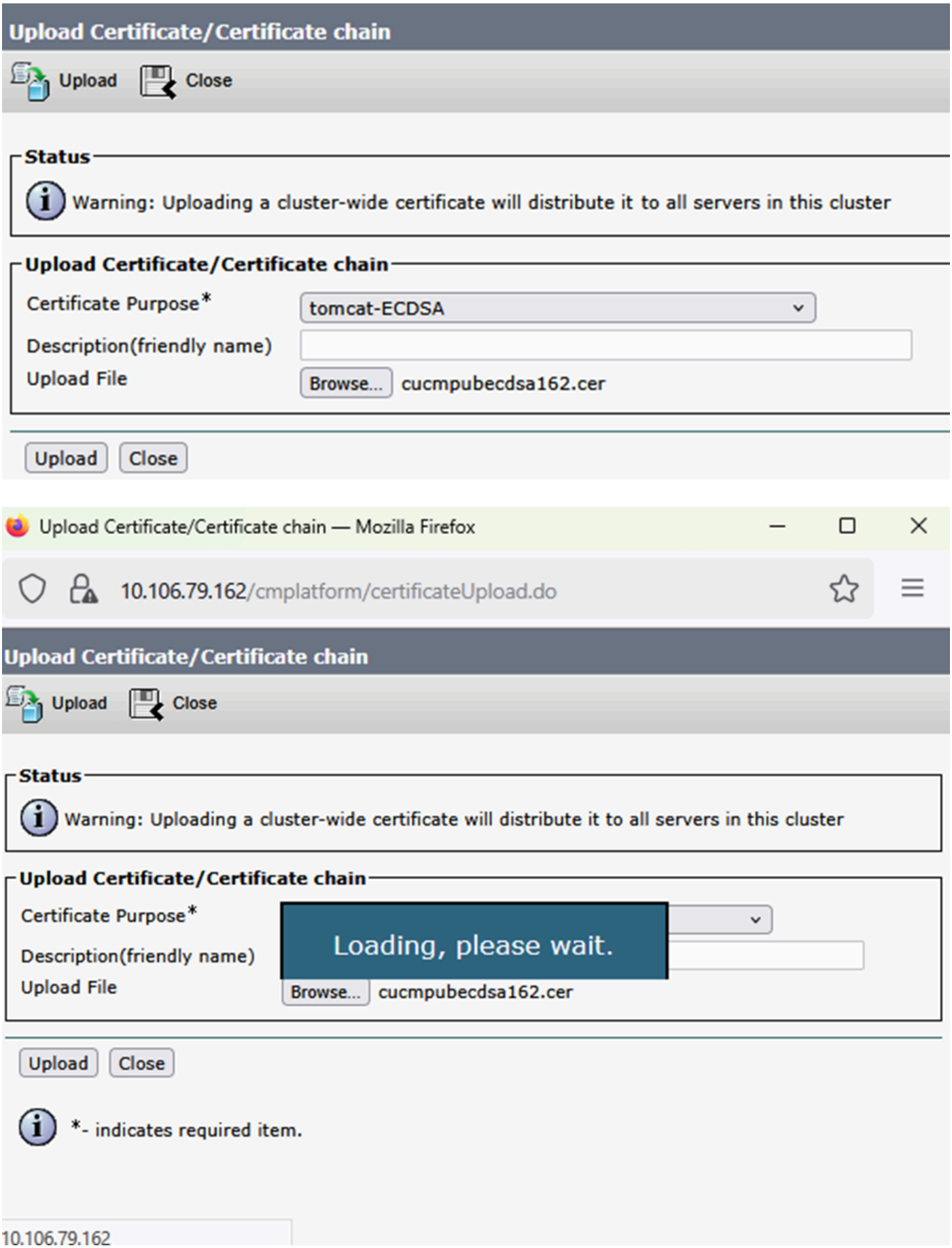
현재 Tomcat-ECDSA 인증서는 자체 서명되어 있습니다.

tomcat	10.106.79.162_5aceb67f00000000000f	Identity	CA-signed	RSA	Multi-server(SAN)	RICKY200-TMS-CA	10/25/2027	Certificate Signed by RICKY200-TMS-CA
tomcat-ECDSA	cucmpubnew-EC.tomcat.com_4b404cd202f547cabf8a6db78c715d46	Identity	self-signed	EC	cucmpubnew.tomcat.com	cucmpubnew-EC.tomcat.com	10/23/2030	self-signed certificate generated by system

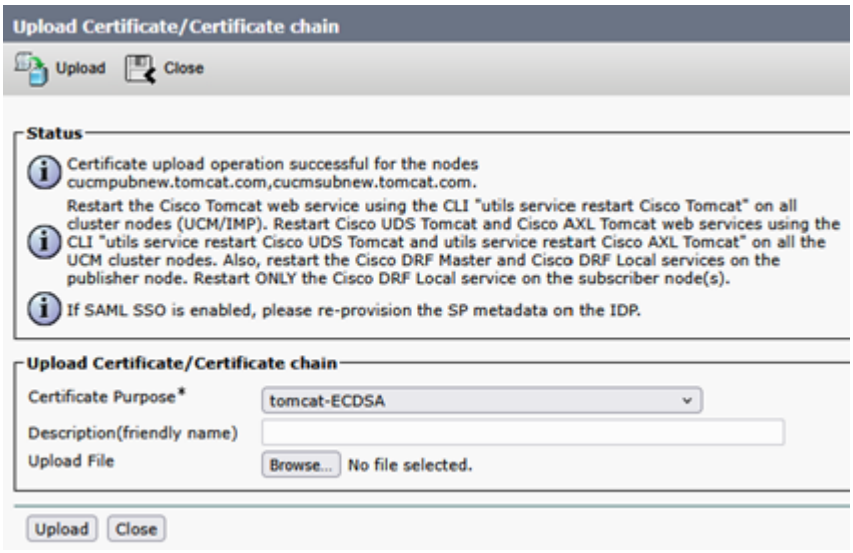
Tomcat-ECDSA 인증서에 대해 다중 CSR에 서명합니다.



CSR을 사용하여 인증서를 서명하고 업로드합니다.

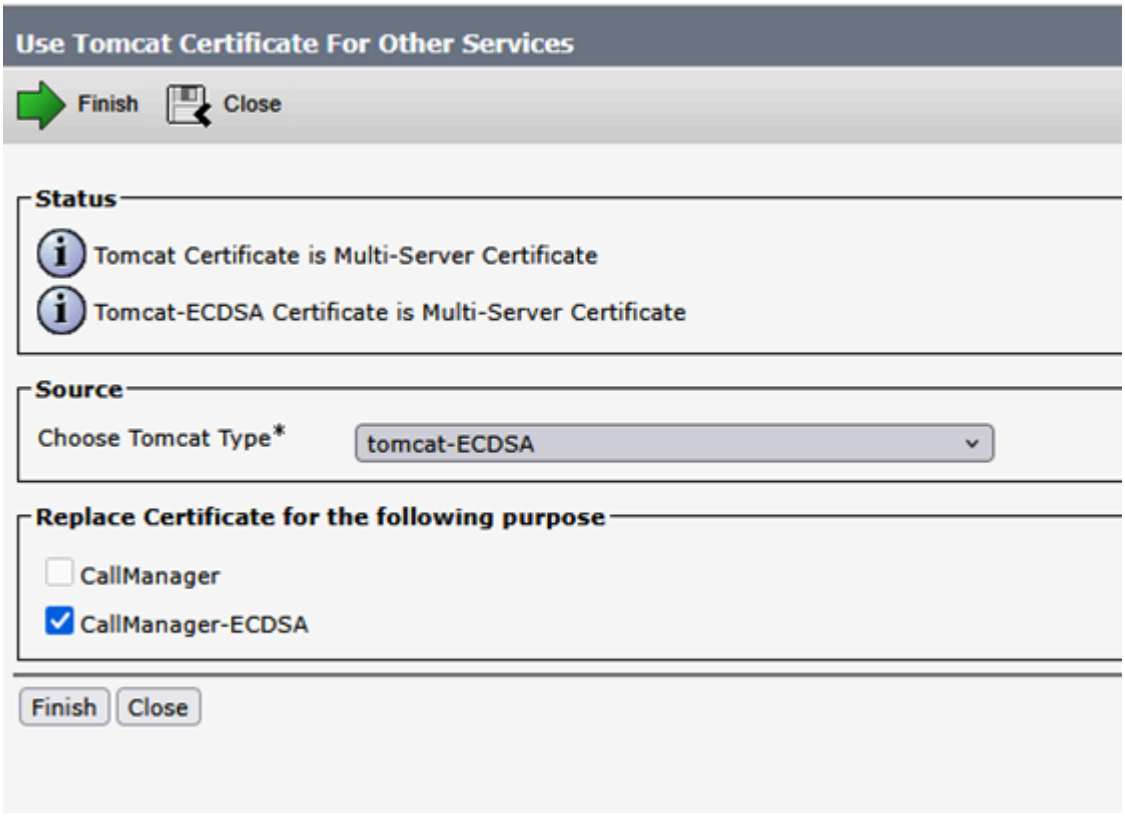


업로드 성공. 프롬프트에 따라 관련 서비스를 다시 시작합니다.

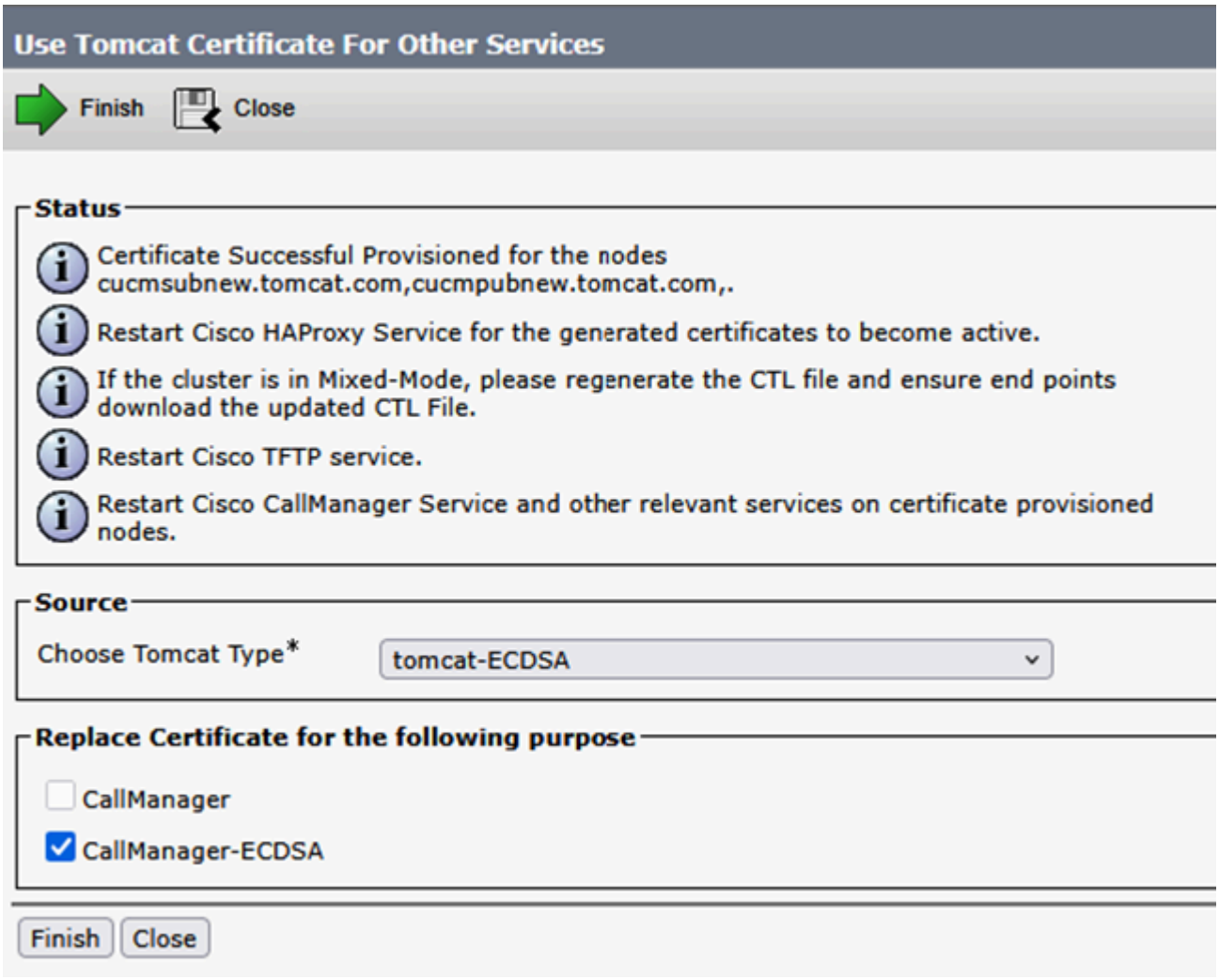


CA에서 서명한 Tomcat 및 Tomcat-ECDSA

이제 Tomcat-ECDSA를 Callmanager-ECDSA 인증서로 다시 사용합니다.



업로드 성공. 프롬프트에 따라 관련 서비스를 다시 시작합니다.



CLI에서 인증서를 확인합니다.

Callmanager-ECDSA 인증서 SN: 2f:00:00:00:38:80:be:cc:a8:a1:8e:8f:23:00:00:00:00:38

```
admin:show cert own CallManager-ecdsa
Invalid Certificate Name. Certificate Not Found.

admin:show cert own CallManager-Ecdsa
Invalid Certificate Name. Certificate Not Found.

admin:show cert own tomcat-ECDSA
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            2f:00:00:00:38:80:be:cc:a8:a1:8e:8f:23:00:00:00:00:00:38
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: DC = com, DC = bgluclab, CN = bgluclab-WIN-DC-01-CA
        Validity
            Not Before: Oct 25 06:46:37 2025 GMT
            Not After : Oct 25 06:46:37 2026 GMT
```

Tomcat-ECDSA 인증서 SN: 2f:00:00:00:38:80:be:cc:a8:a1:8e:8f:23:00:00:00:00:00:38.

```
admin:show cert own tomcat-ECDSA
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            2f:00:00:00:38:80:be:cc:a8:a1:8e:8f:23:00:00:00:00:00:38
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: DC = com, DC = bgluclab, CN = bgluclab-WIN-DC-01-CA
        Validity
            Not Before: Oct 25 06:46:37 2025 GMT
            Not After : Oct 25 06:46:37 2026 GMT
        Subject: C = IN, ST = karnataka, L = bgl, O = cisco, CN = cucmpubnew-EC-ms.tomcat.com
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
```

이제 두 서비스에 하나의 인증서, 즉 Tomcat 및 Callmanager 서비스용 Tomcat 인증서와 Tomcat-ECDSA 및 Callmanager-ECDSA 서비스용 Tomcat-ECDSA를 사용하고 있으므로 Expressway Trust Store에서 인증서를 업로드하는 것이 더 번거로워집니다(업로드해야 하는 경우).

MRA용 Expressway-Core에서 UCM을 추가하는 동안 TLS에서 'On'을 확인하도록 하는 것이 그 어느 때보다 쉬워졌습니다. Tomcat 인증서 CA 또는 서버 인증서를 하나만 추가하면 작업이 수행됩니다(인증서가 Callmanager와 Tomcat 서비스 간에 공유되기 때문).

x14.2 이상으로 업그레이드하여 Mobile Remote Access를 중단한 경우 이 포괄적인 문서를 참조하여 문제를 해결할 수도 있습니다.

Apache Traffic Server 버전 기록

루트에 로그인한 서버의 버전을 확인하고 ~ # /apache2/bin/httpd -v를 실행합니다.

Expressway x8.11.4
서버 버전: Apache/2.4.34(Unix)
서버 구축: 2018년 11월 12일 19:04:23

Expressway x12.6

서버 버전: Apache/2.4.43(Unix)
서버 구축: 2020년 5월 26일 18:27:21

Expressway x14.0.8

서버 버전: Apache/2.4.53(Unix)
서버 구축: 2022년 5월 4일 08:52:57

Expressway x15.3

서버 버전: Apache/2.4.62(Unix)
서버 구축: 2025년 7월 16일 12:10:19

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.