ISE를 사용하는 IOS XE 디바이스에서 TLS 1.3을 통해 TACACS+ 구성

목차

소개

이 문서에서는 Cisco ISE(Identity Services Engine)를 서버로, Cisco IOS® XE 디바이스를 클라이언트로 사용하는 TACACS+ over TLS의 예를 설명합니다.

개요

TACACS+(Terminal Access Controller Access-Control System Plus) 프로토콜[RFC8907]은 하나 이상의 TACACS+ 서버를 통해 라우터, 네트워크 액세스 서버 및 기타 네트워크 디바이스에 대한 중앙 집중식 디바이스 관리를 가능하게 합니다. 특히 디바이스 관리 활용 사례에 맞게 맞춤화된 AAA(Authentication, Authorization, and Accounting) 서비스를 제공합니다.

TACACS+ over TLS 1.3 [RFC8446]은 보안 전송 계층을 도입하여 프로토콜을 개선하고 매우 중요한 데이터를 보호합니다. 이러한 통합으로 TACACS+ 클라이언트와 서버 간의 연결 및 네트워크 트래픽에 대한 기밀성, 무결성 및 인증이 보장됩니다.

설명서 사용

이 설명서는 ISE가 Cisco IOS XE 기반 네트워크 디바이스에 대한 관리 액세스를 관리할 수 있도록 활동을 두 부분으로 나눕니다.
· 1부 - 디바이스 관리를 위한 ISE 구성
· 2부 - TACACS+ over TLS용 Cisco IOS XE 구성

사전 요구 사항

요구 사항

TLS를 통해 TACACS+를 구성하기 위한 요구 사항:

- TACACS+에서 TLS를 통해 ISE 및 네트워크 디바이스의 인증서를 서명하는 데 사용되는 인증서를 서명하는 CA(Certificate Authority).
- CA(Certificate Authority)의 루트 인증서
- 네트워크 디바이스 및 ISE는 DNS 연결성을 가지며 호스트 이름을 확인할 수 있습니다.

사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- ISE VMware 가상 어플라이언스, 릴리스 3.4 패치 2
- Cisco IOS XE Software, 버전 17.15+

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

라이센싱

Device Administration 라이센스를 사용하면 정책 서비스 노드에서 TACACS+ 서비스를 사용할 수 있습니다. 고가용성(HA) 독립형 구축에서 디바이스 관리 라이센스를 사용하면 HA 쌍의 단일 정책 서비스 노드에서 TACACS+ 서비스를 사용할 수 있습니다.

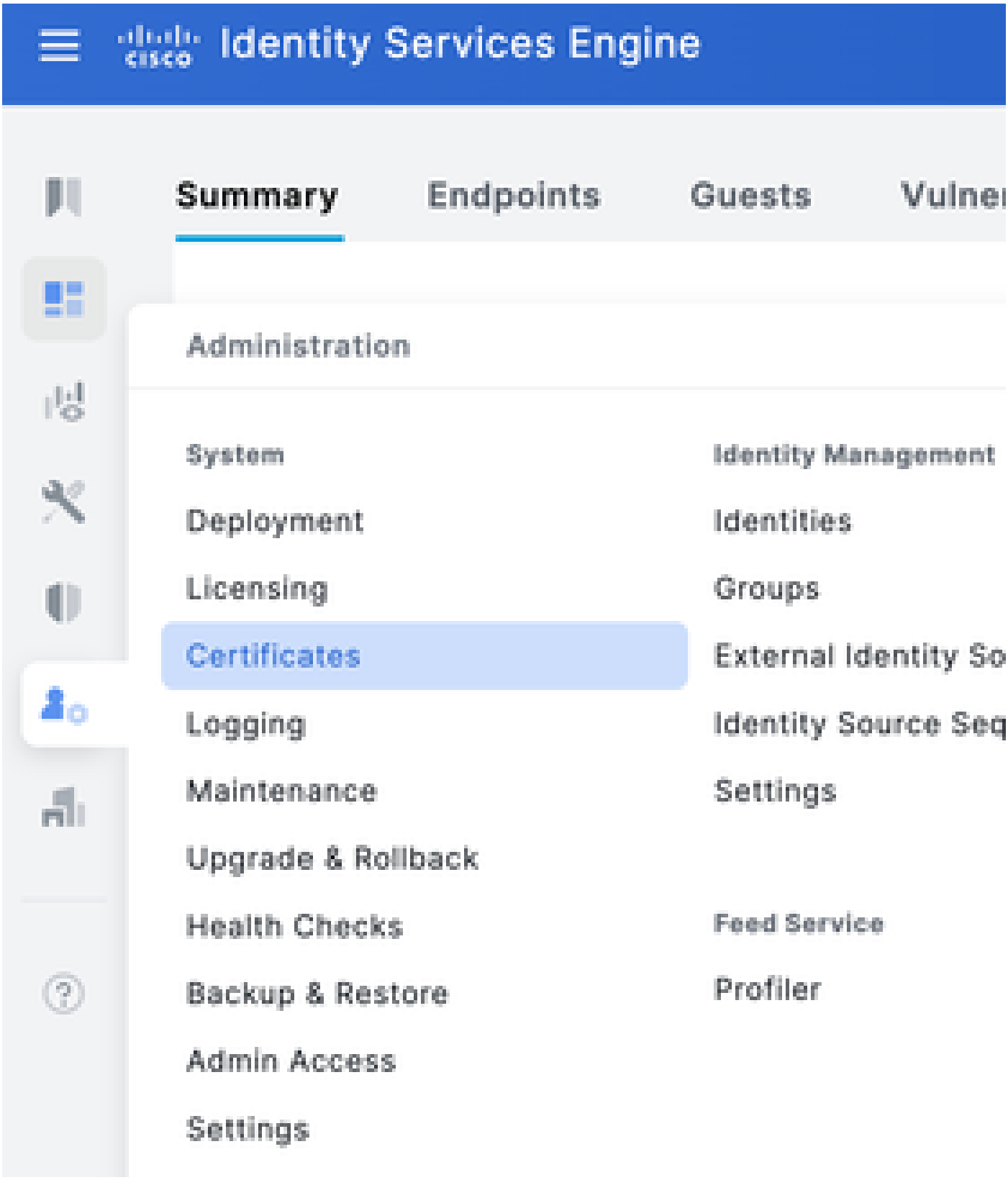
1부 - 디바이스 관리를 위한 ISE 구성

TACACS+ 서버 인증을 위한 인증서 서명 요청 생성

1단계. 지원되는 브라우저 중 하나를 사용하여 ISE 관리 웹 포털에 로그인합니다.

기본적으로 ISE는 모든 서비스에 자체 서명 인증서를 사용합니다. 첫 번째 단계는 CSR(Certificate Signing Request)을 생성하여 CA(Certificate Authority)에서 서명하도록 하는 것입니다.

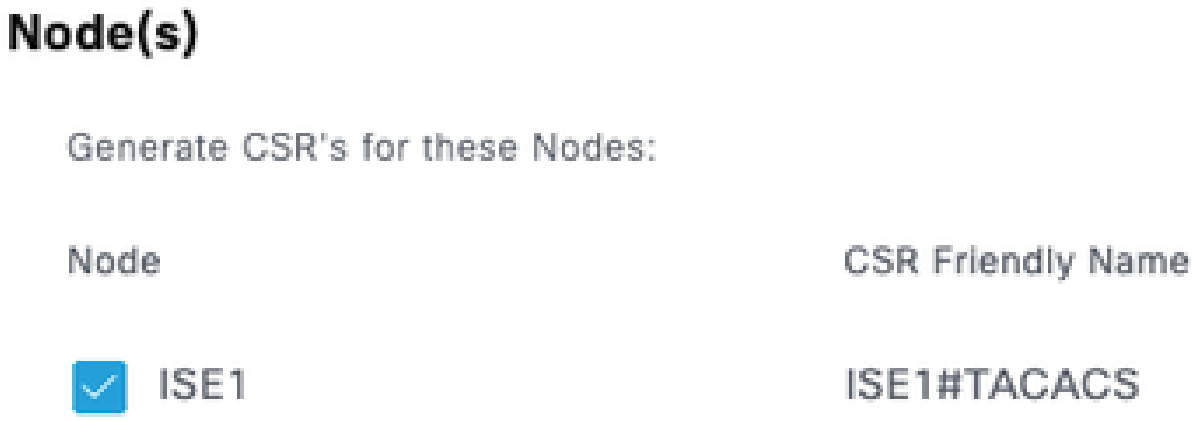
2단계. Administration(관리) > System(시스템) > Certificates(인증서)로 이동합니다.



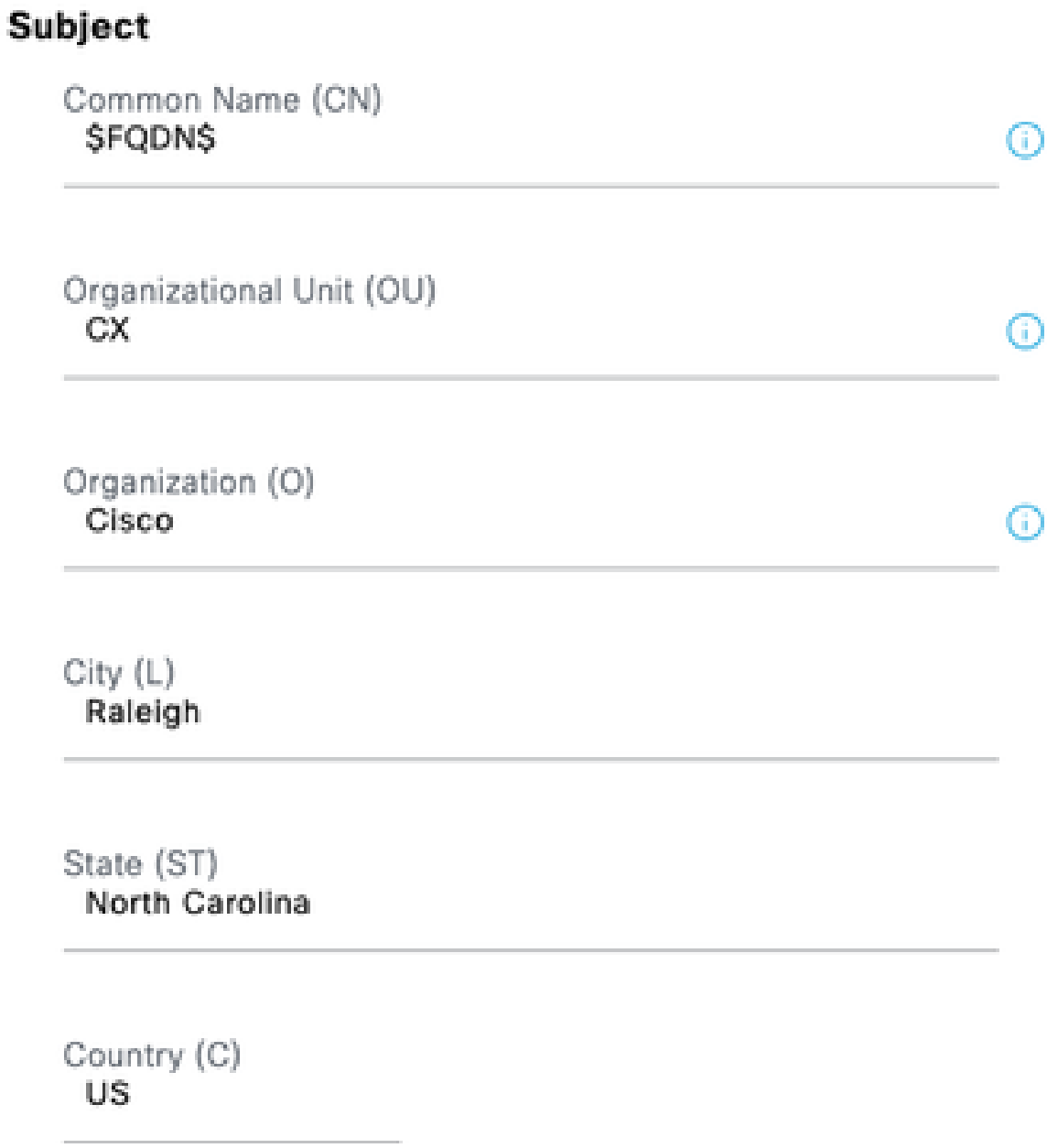
3단계. Certificate Signing Requests(인증서 서명 요청) 아래에서 Generate Certificate Signing Request(인증서 서명 요청 생성)를 클릭합니다.

4단계. TACACS inUsage를 선택합니다.

5단계. TACACS+를 활성화할 PSN을 선택합니다.

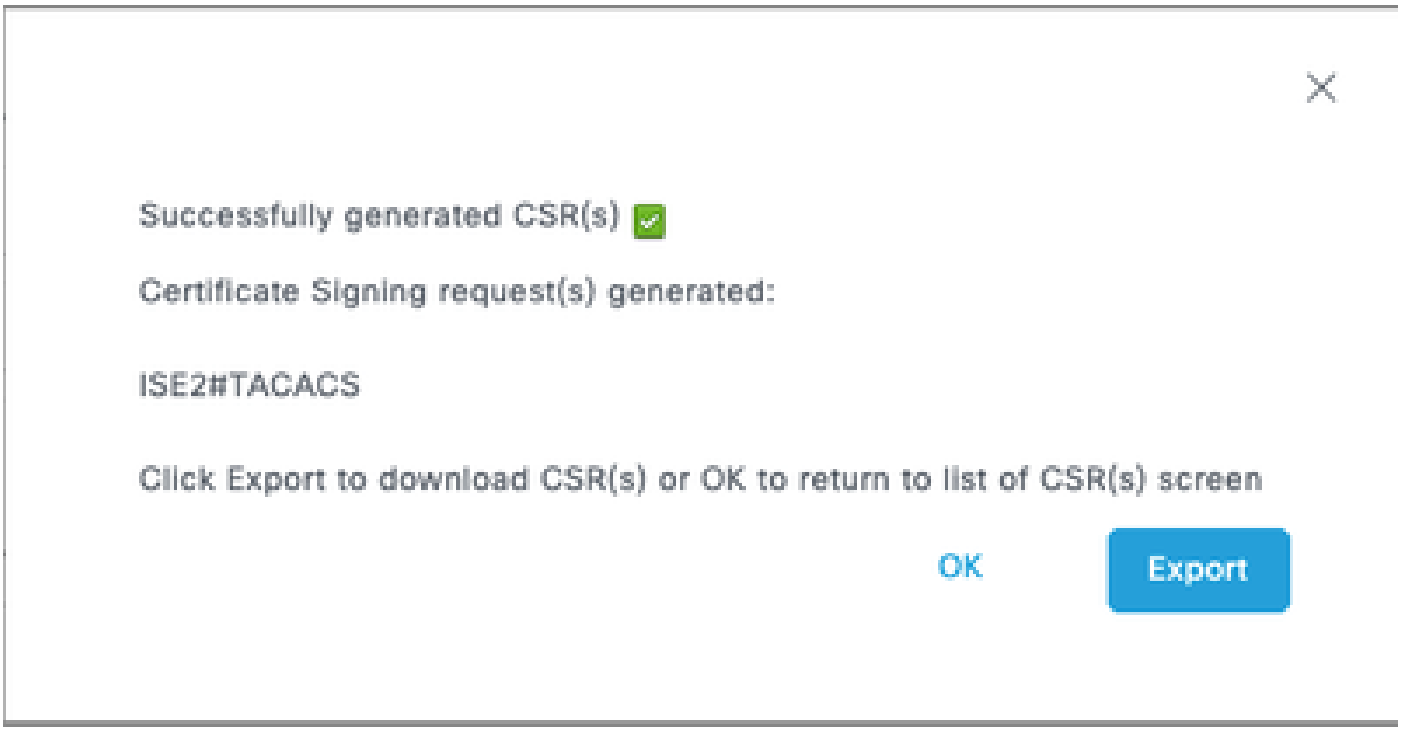


6단계. Subject(제목) 필드에 적절한 정보를 입력합니다.



7단계. SAN(Subject Alternative Name) 아래에 DNS 이름 및 IP 주소를 추가합니다.

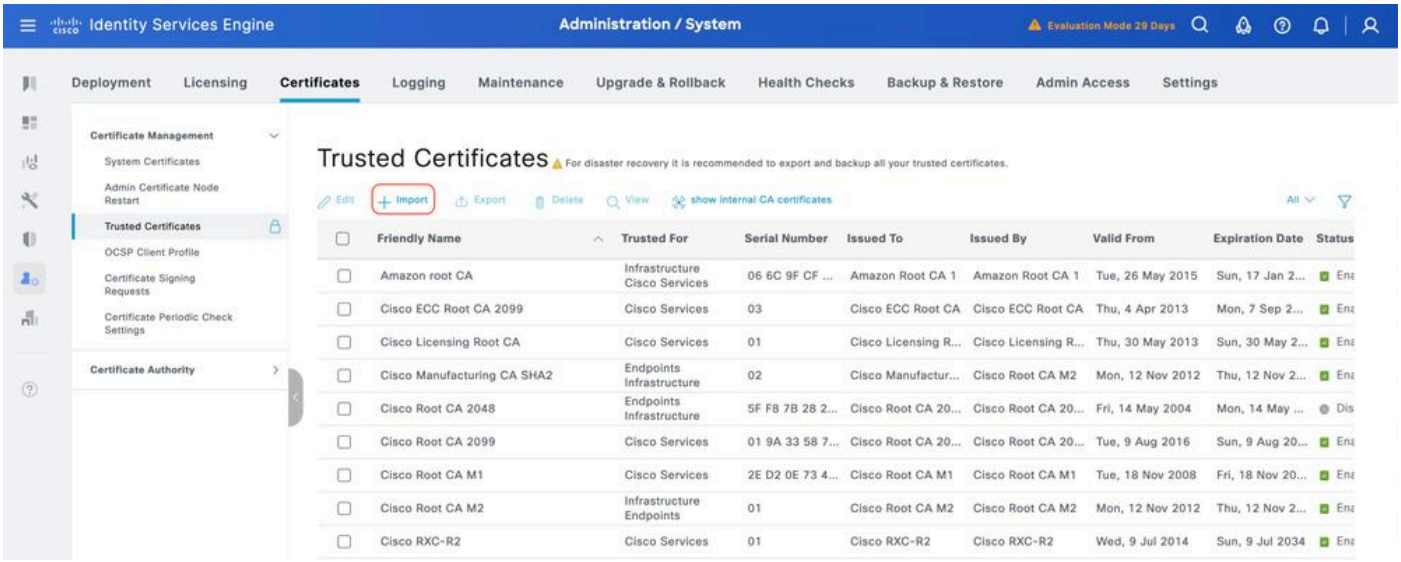
8단계. Generate(생성)를 클릭한 다음 Export(내보내기)를 클릭합니다.



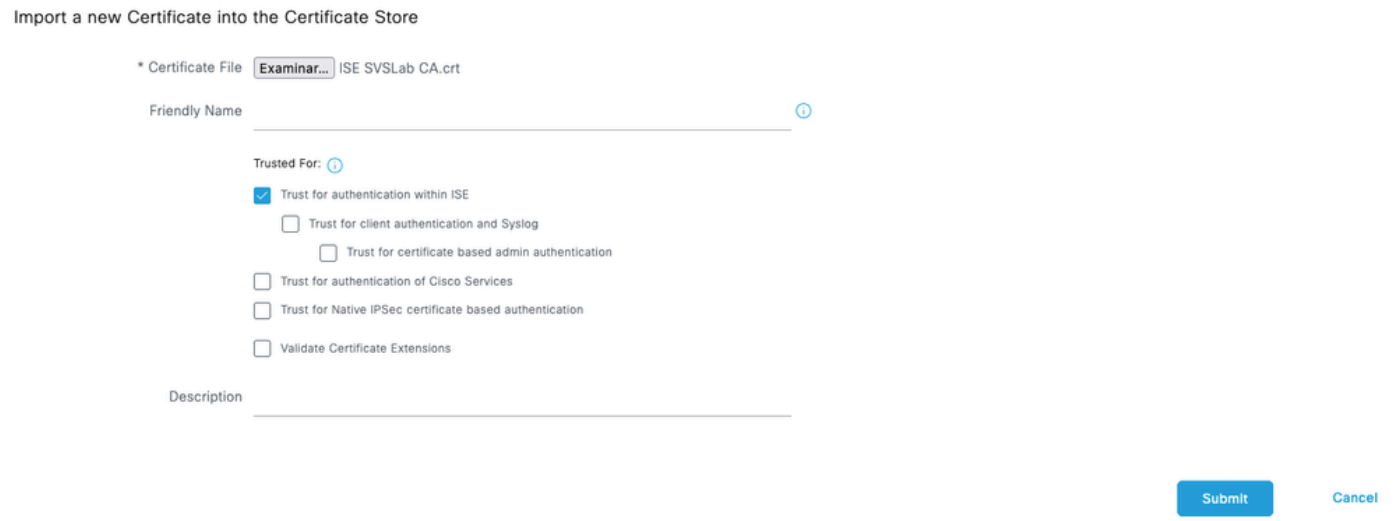
이제 CA(Certificate Authority)에서 서명한 인증서(CRT)를 사용할 수 있습니다.

TACACS+ 서버 인증을 위한 루트 CA 인증서 업로드

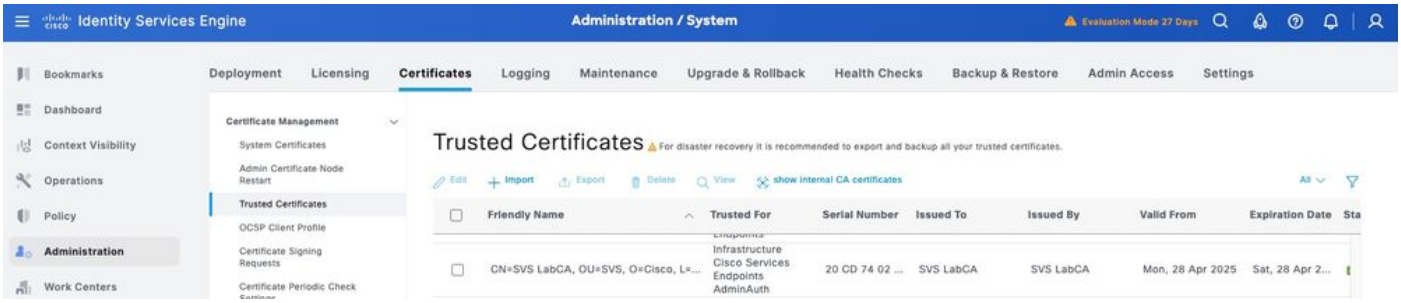
1단계. Administration(관리) > System(시스템) > Certificates(인증서)로 이동합니다. Trusted Certificates(신뢰할 수 있는 인증서) 아래에서 Import(가져오기)를 클릭합니다.



2단계. TACACS CSR(Certificate Signing Request)에 서명한 CA(Certificate Authority)에서 발급한 인증서를 선택합니다. 다음 사항을 확인하십시오.ISE 내 인증 신뢰 옵션을 사용할 수 있습니다.



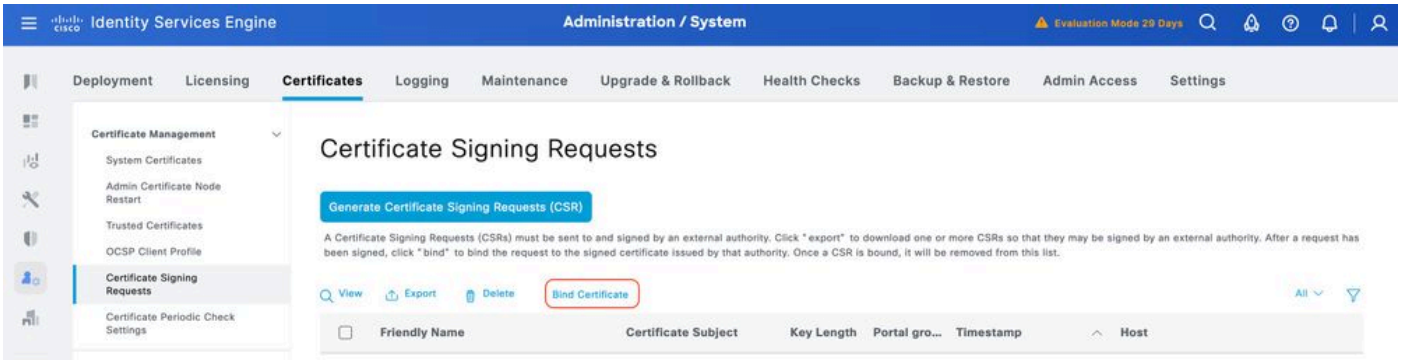
3단계. Submit(제출)을 클릭합니다. 이제 인증서가 Trusted Certificates(신뢰할 수 있는 인증서) 아래에 나타나야 합니다.



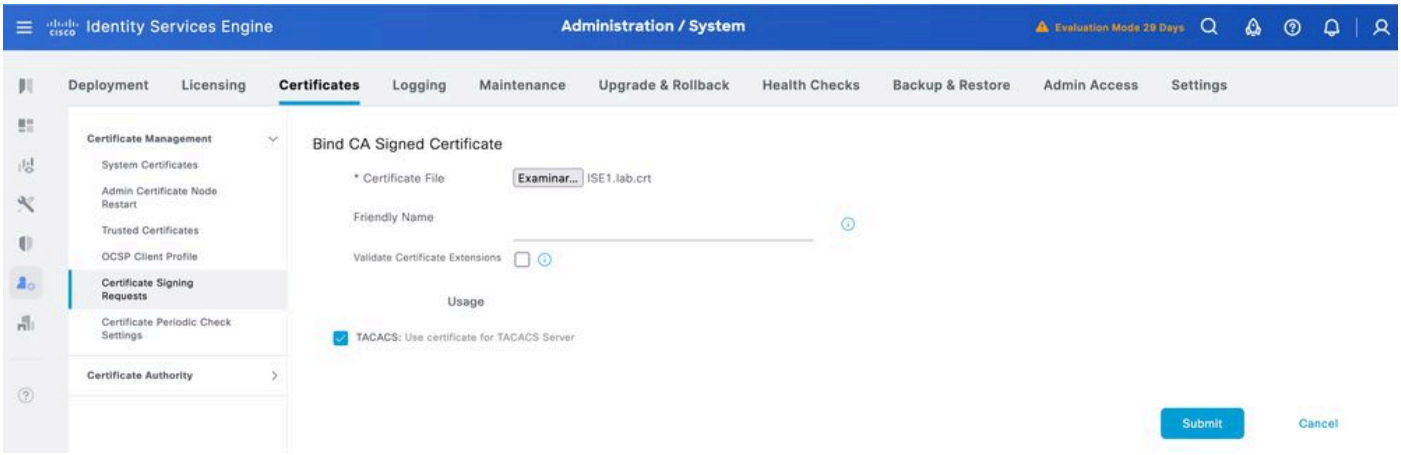
ISE에 서명된 CSR(Certificate Signing Request) 바인딩

CSR(Certificate Signing Request)에 서명하면 ISE에 서명된 인증서를 설치할 수 있습니다.

1단계. Administration(관리) > System(시스템) > Certificates(인증서)로 이동합니다. Certificate Signing Requests(인증서 서명 요청) 아래에서 이전 단계에서 생성된 TACACS CSR을 선택하고 Bind Certificate(인증서 바인딩)를 클릭합니다.

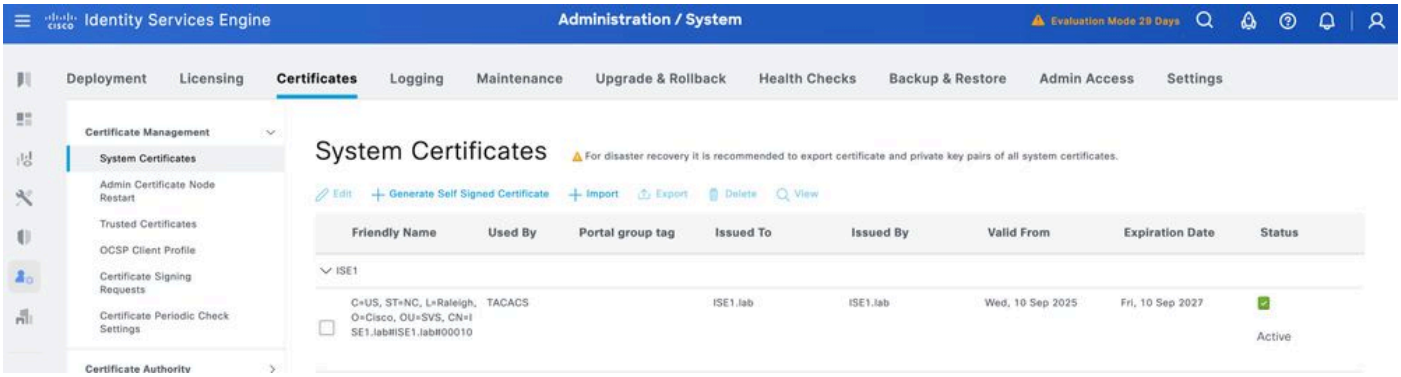


2단계. 서명된 인증서를 선택하고 Usage(사용) 아래의 TACACS(TACACS) 확인란이 선택된 상태로 유지되는지 확인합니다.



3단계. Submit(제출)을 클릭합니다. 기존 인증서 교체에 대한 경고가 표시되면 Yes(예)를 클릭하여 계속 진행합니다.

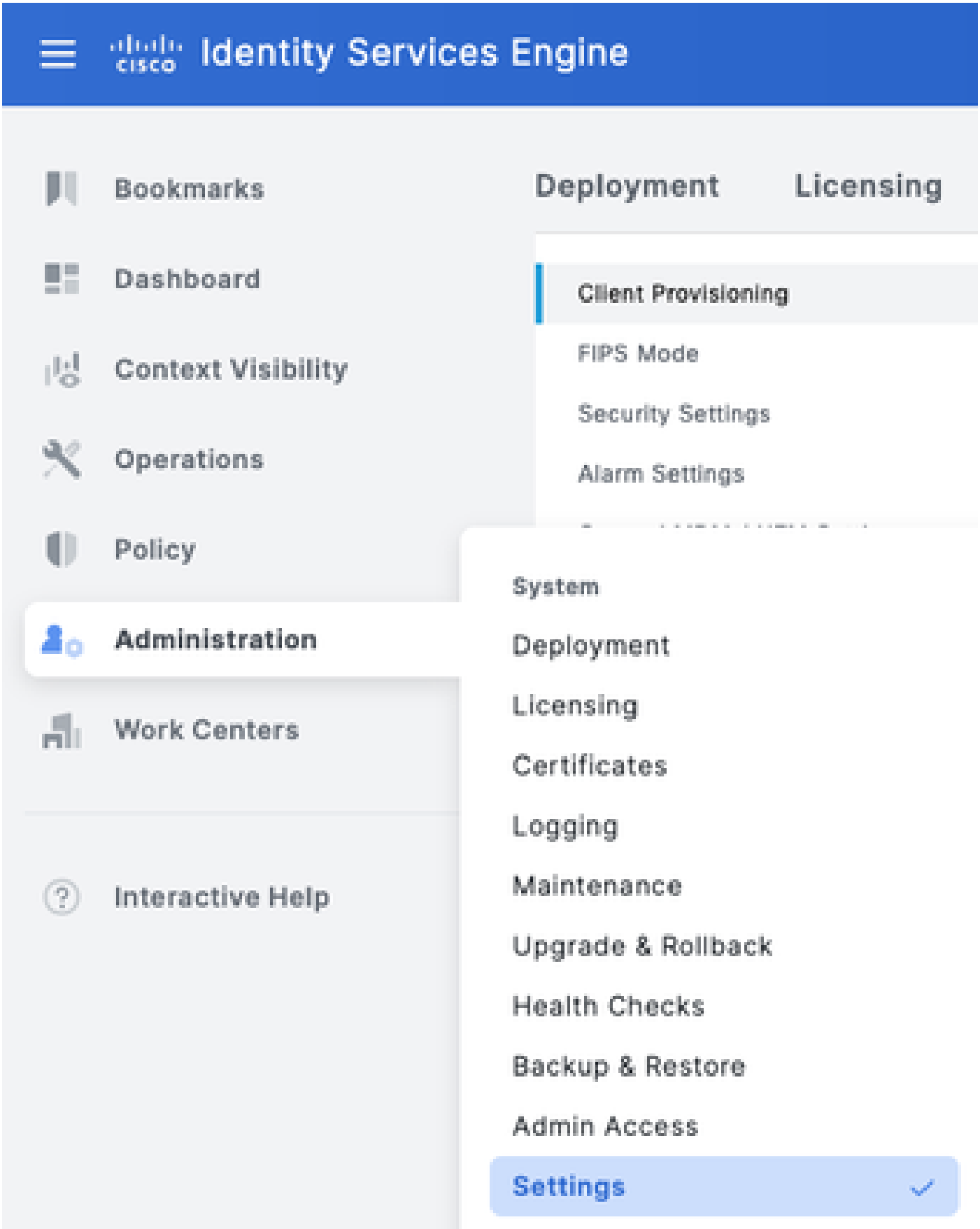
이제 인증서가 올바르게 설치되어야 합니다. System Certificates(시스템 인증서)에서 이를 확인할 수 있습니다.



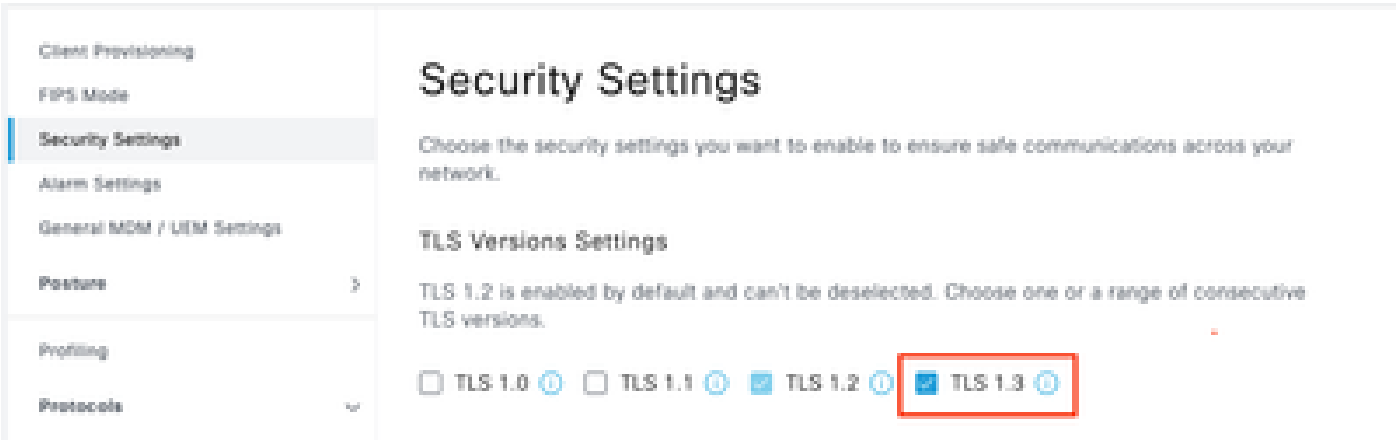
TLS 1.3 활성화

TLS 1.3은 ISE 3.4.x에서 기본적으로 활성화되지 않습니다. 수동으로 활성화해야 합니다.

1단계. Administration(관리) > System(시스템) > Settings(설정)로 이동합니다.



2단계. Security Settings(보안 설정)를 클릭하고 TLS Version Settings(TLS 버전 설정)에서 TLS1.3 옆에 있는 확인란을 선택한 다음 Save(저장)를 클릭합니다.

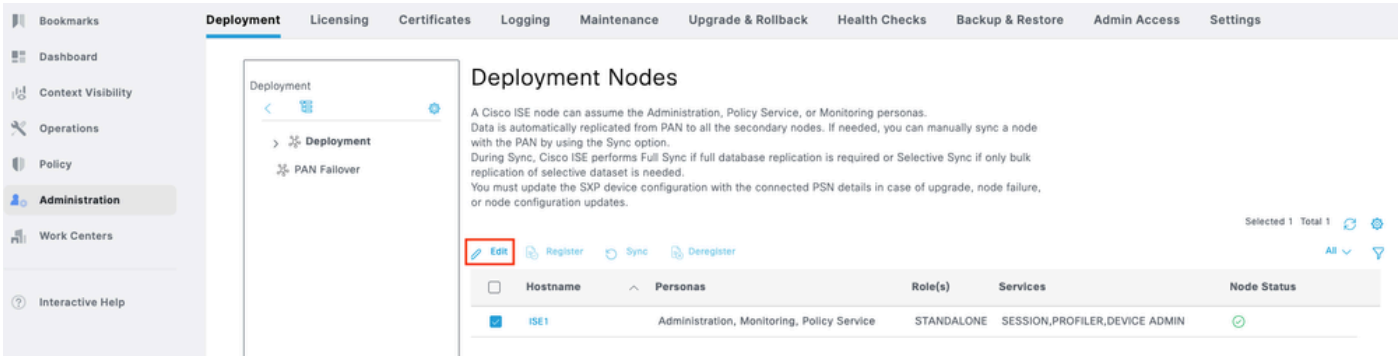


경고: TLS 버전을 변경하면 Cisco ISE 애플리케이션 서버가 모든 Cisco ISE 구축 시스템에서 다시 시작됩니다.

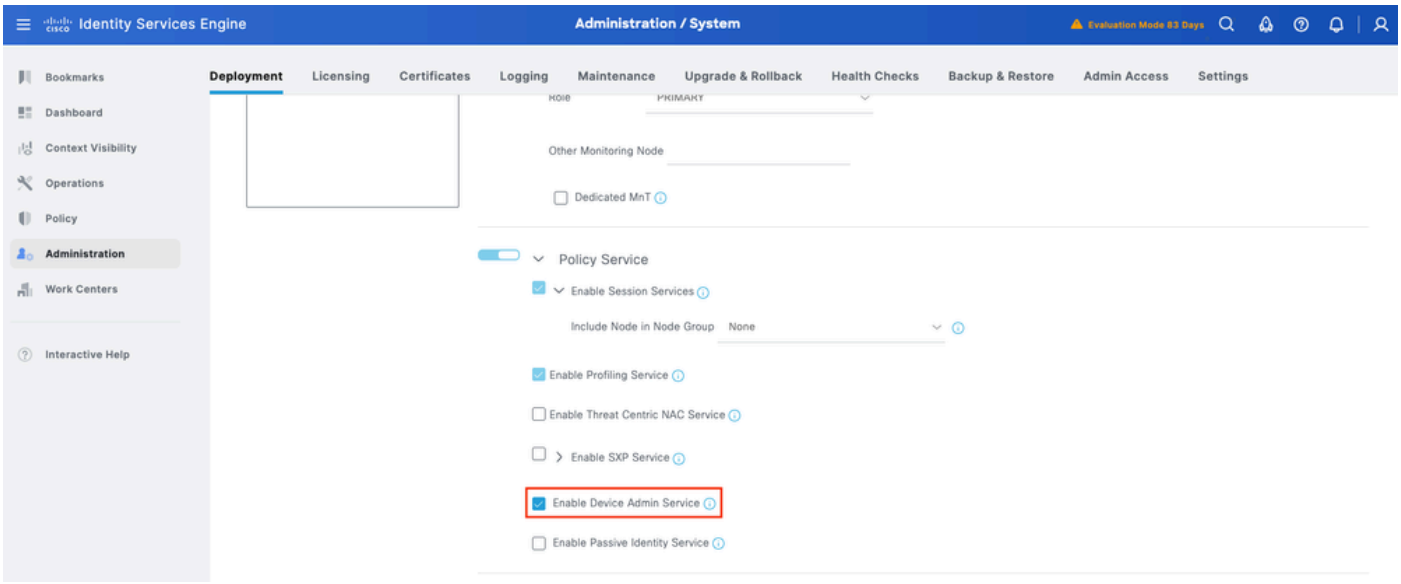
ISE에서 디바이스 관리 활성화

디바이스 관리 서비스(TACACS+)는 ISE 노드에서 기본적으로 활성화되지 않습니다. PSN 노드에서 TACACS+를 활성화합니다.

1단계. Administration(관리) > System(시스템) > Deployment(구축)로 이동합니다. ISE 노드 옆의 확인란을 선택하고 Edit를 클릭합니다.



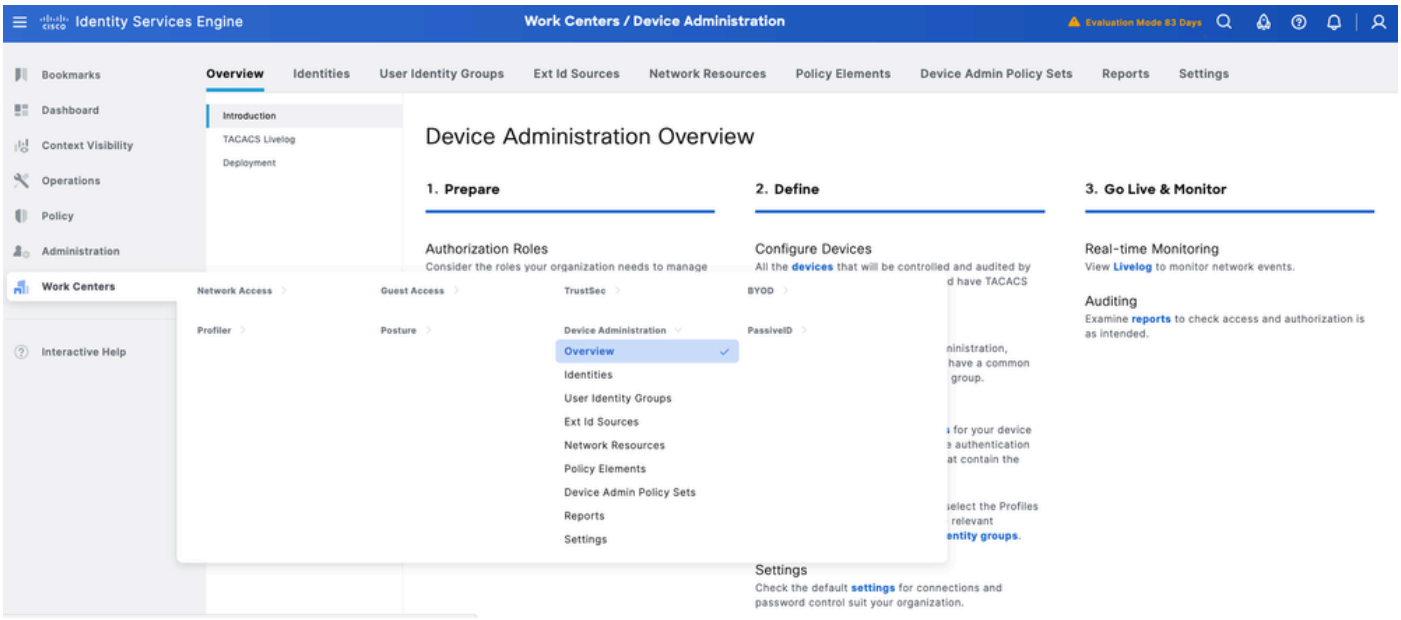
2단계. GeneralSettings(일반 설정)에서 아래로 스크롤하여 Enable Device Admin Service(디바이스 관리 서비스 활성화) 옆에 있는 확인란을 선택합니다.



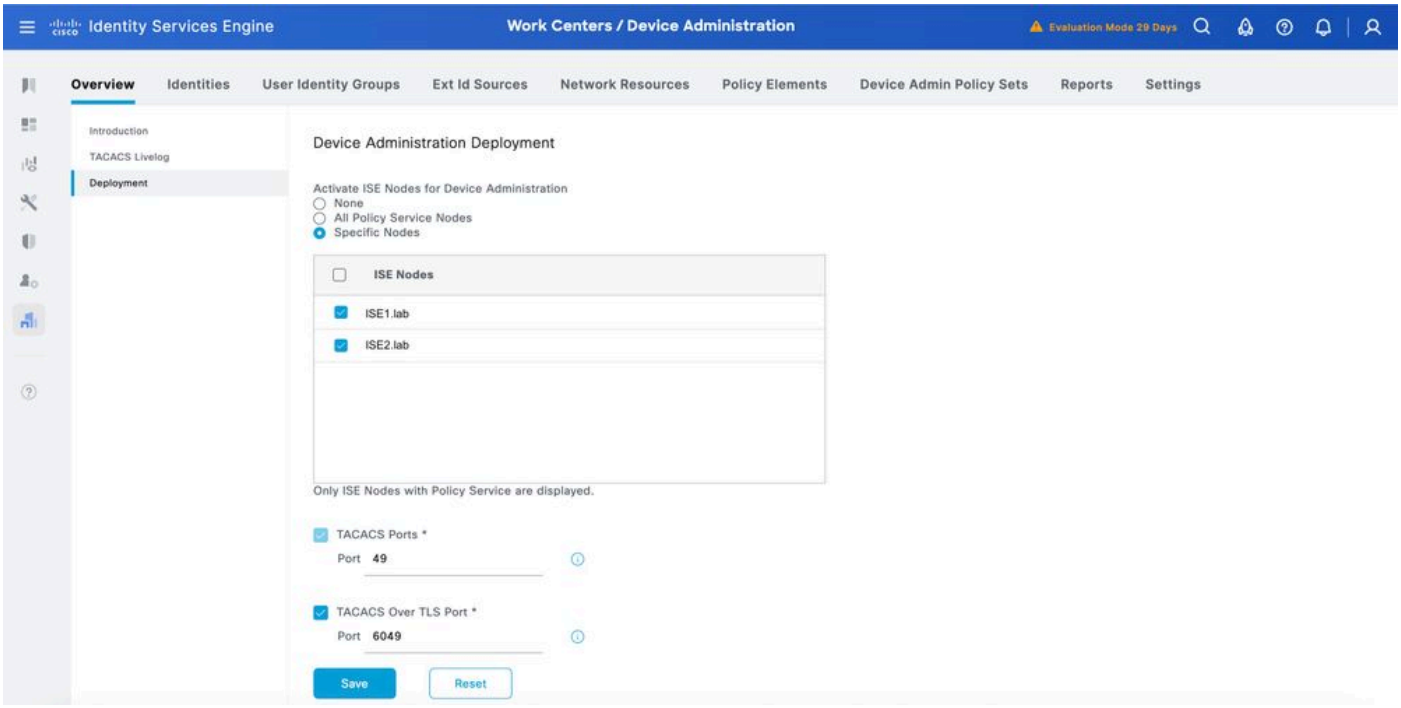
3단계. 구성을 저장합니다. 이제 ISE에서 디바이스 관리 서비스가 활성화됩니다.

TACACS Over TLS 활성화

1단계. Work Centers(작업 센터) > Device Administration(디바이스 관리) > Overview(개요)로 이동합니다.



2단계. Deployment(구축)를 클릭합니다. TACACS over TLS를 활성화할 PSN 노드를 선택합니다.

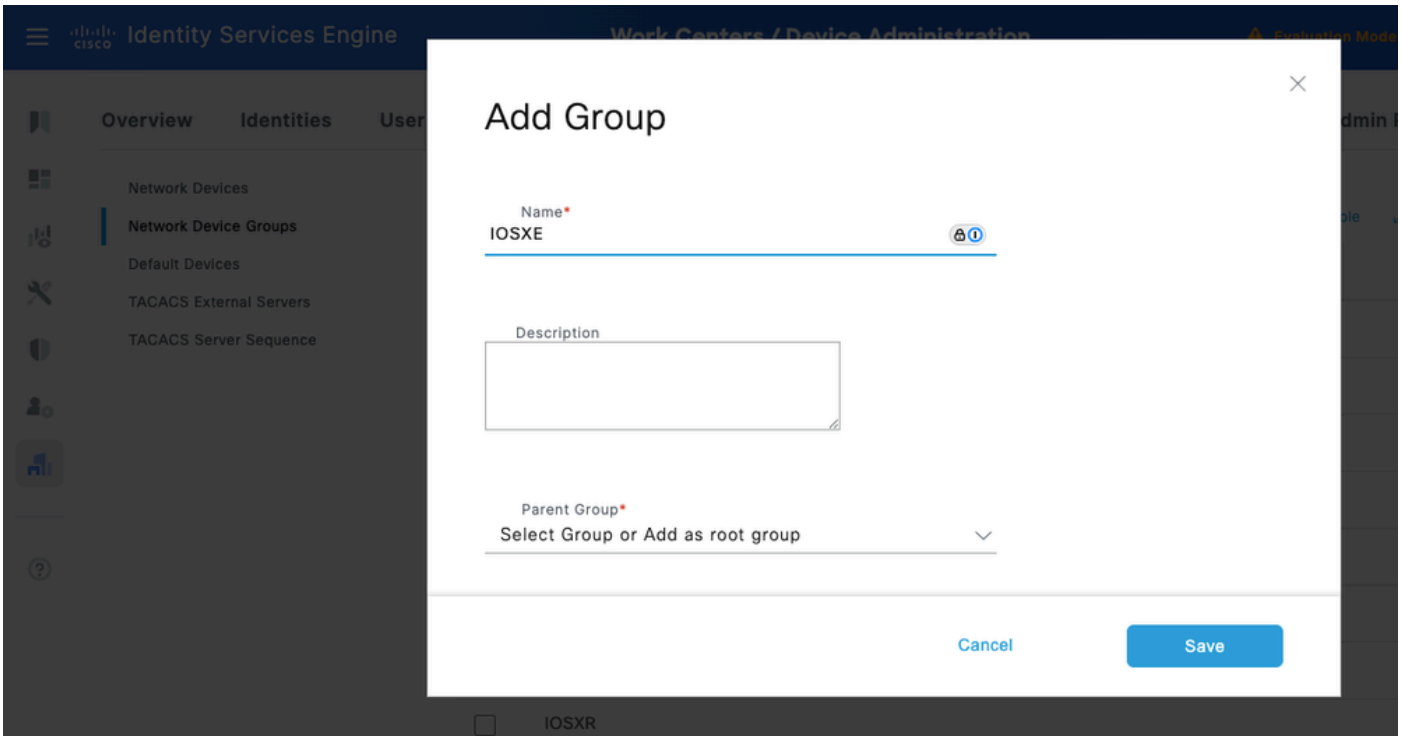


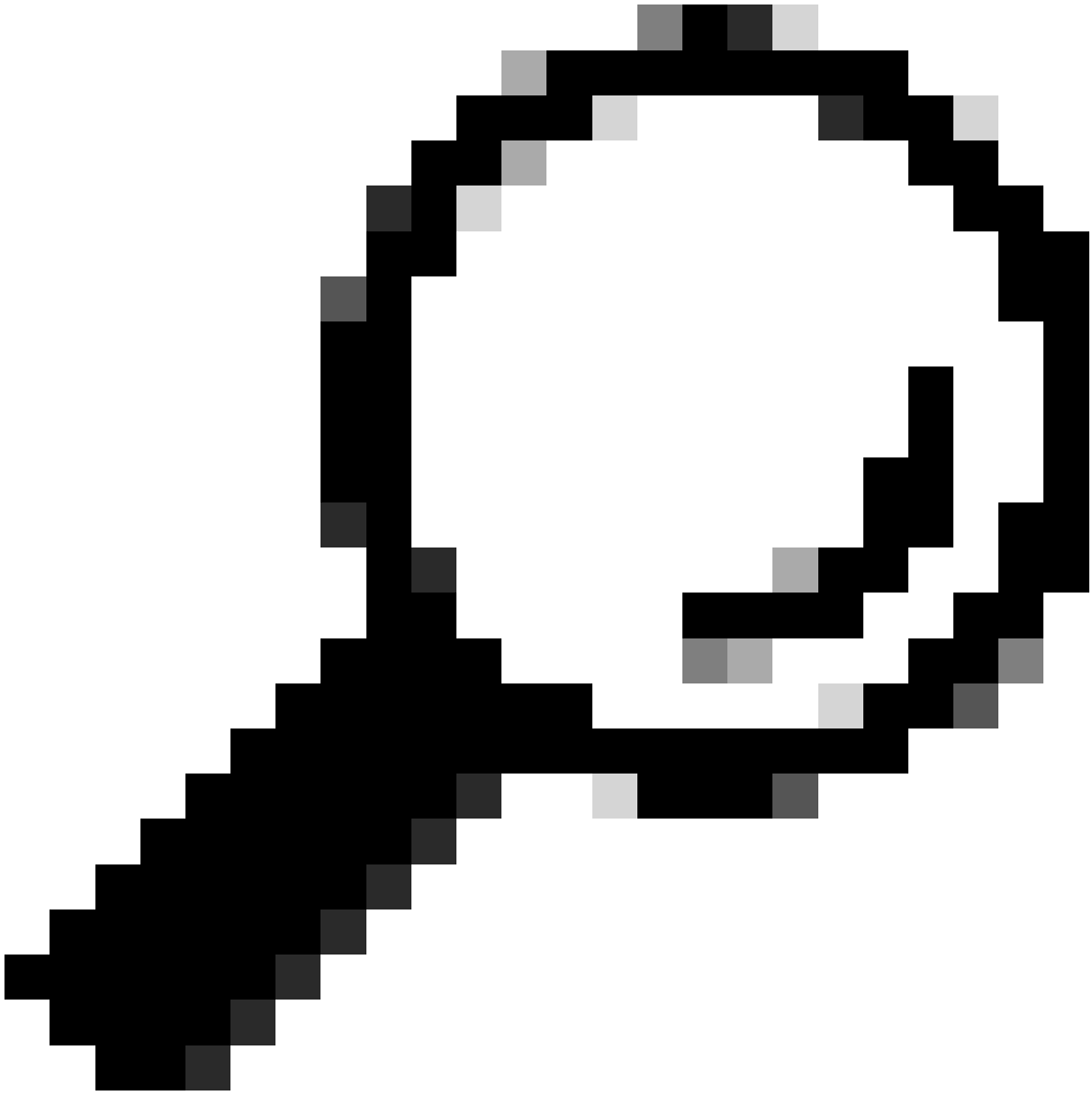
3단계. 기본 포트 6049를 유지하거나 TACACS over TLS에 대해 다른 TCP 포트를 지정한 다음 Save를 클릭합니다.

네트워크 디바이스 및 네트워크 디바이스 그룹 생성

ISE는 여러 디바이스 그룹 계층을 사용하여 강력한 디바이스 그룹화를 제공합니다. 각 계층 구조는 네트워크 장치의 개별적이고 독립적인 분류를 나타냅니다.

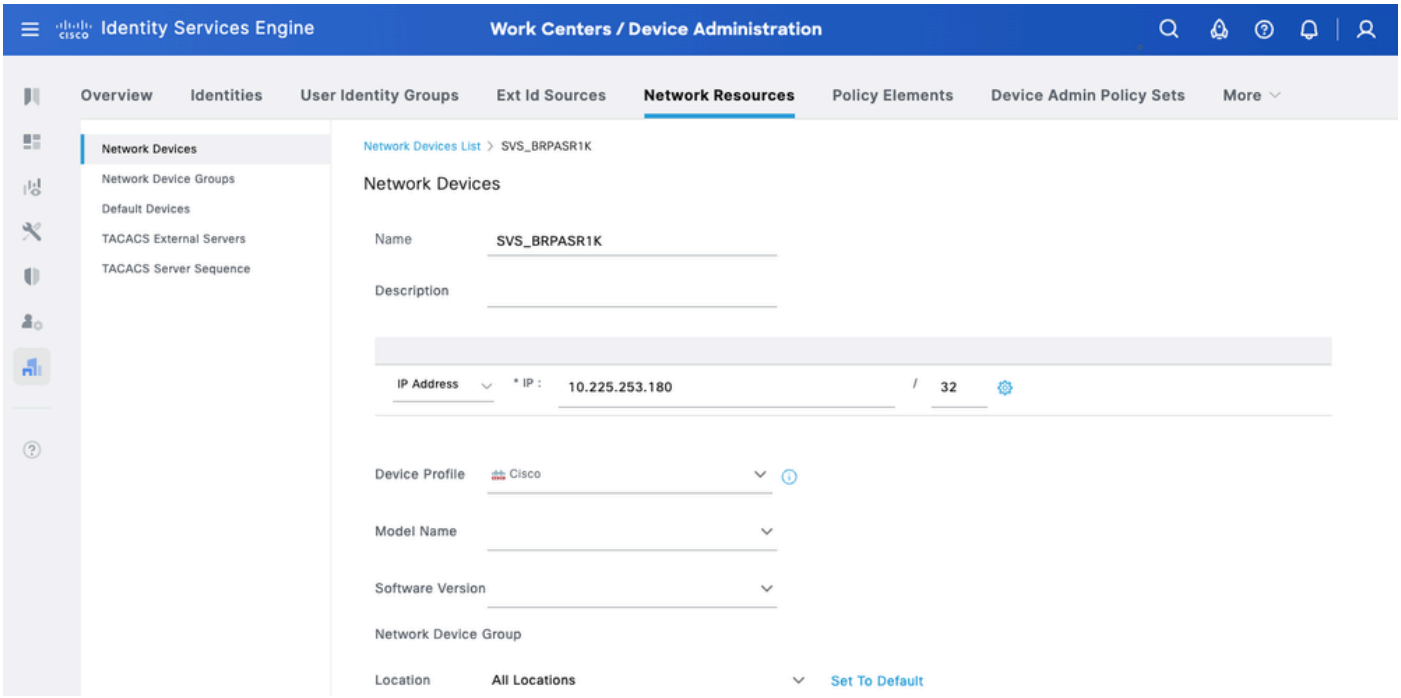
1단계. Work Centers(작업 센터) > Device Administration(디바이스 관리) > Network Resources(네트워크 리소스)로 이동합니다.Network Device Groups(네트워크 디바이스 그룹)를 클릭하고 이름이 IOS XE인 그룹을 생성합니다.


Identity Services Engine
Overview
Identities
Network Devices
Network Device Groups
Default Devices
TACACS External Servers
TACACS Server Sequence
Add Group
Name*
IOSXE
Description
Parent Group*
Select Group or Add as root group
Cancel
Save
IOSXR

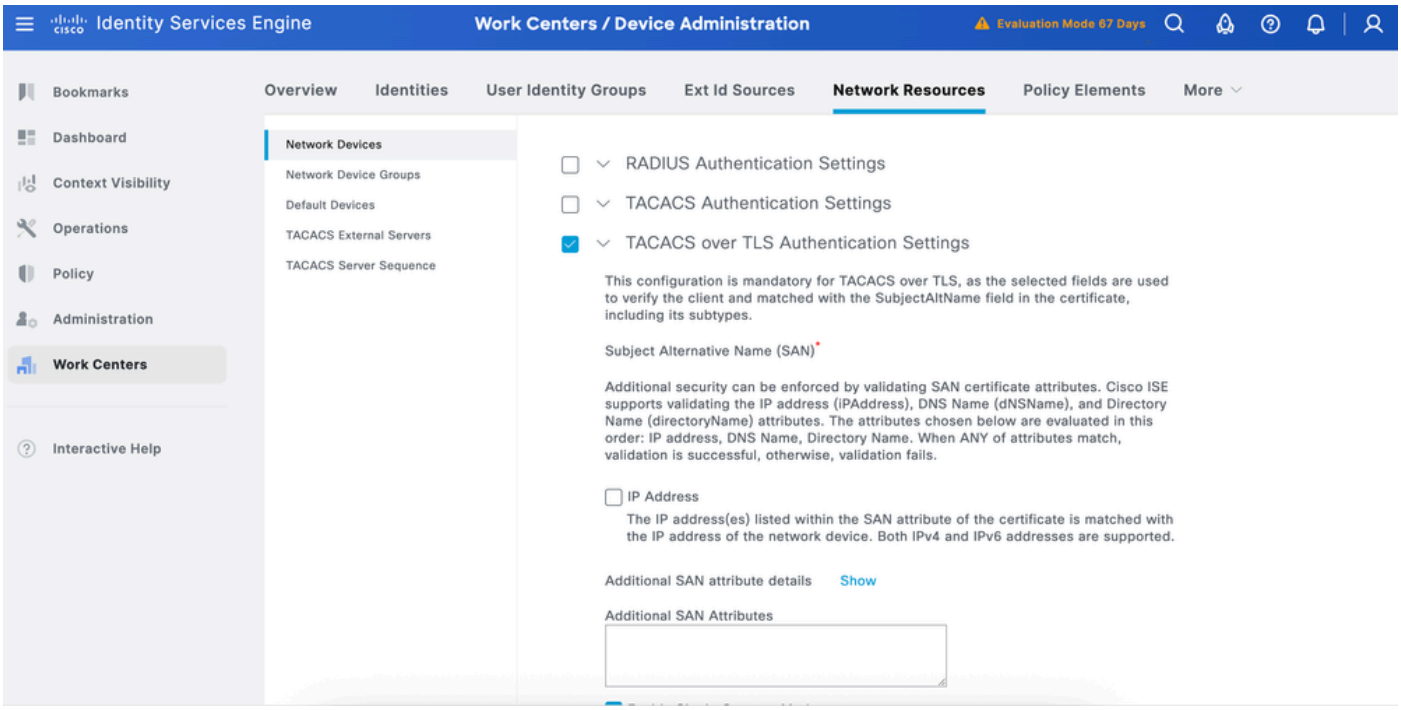


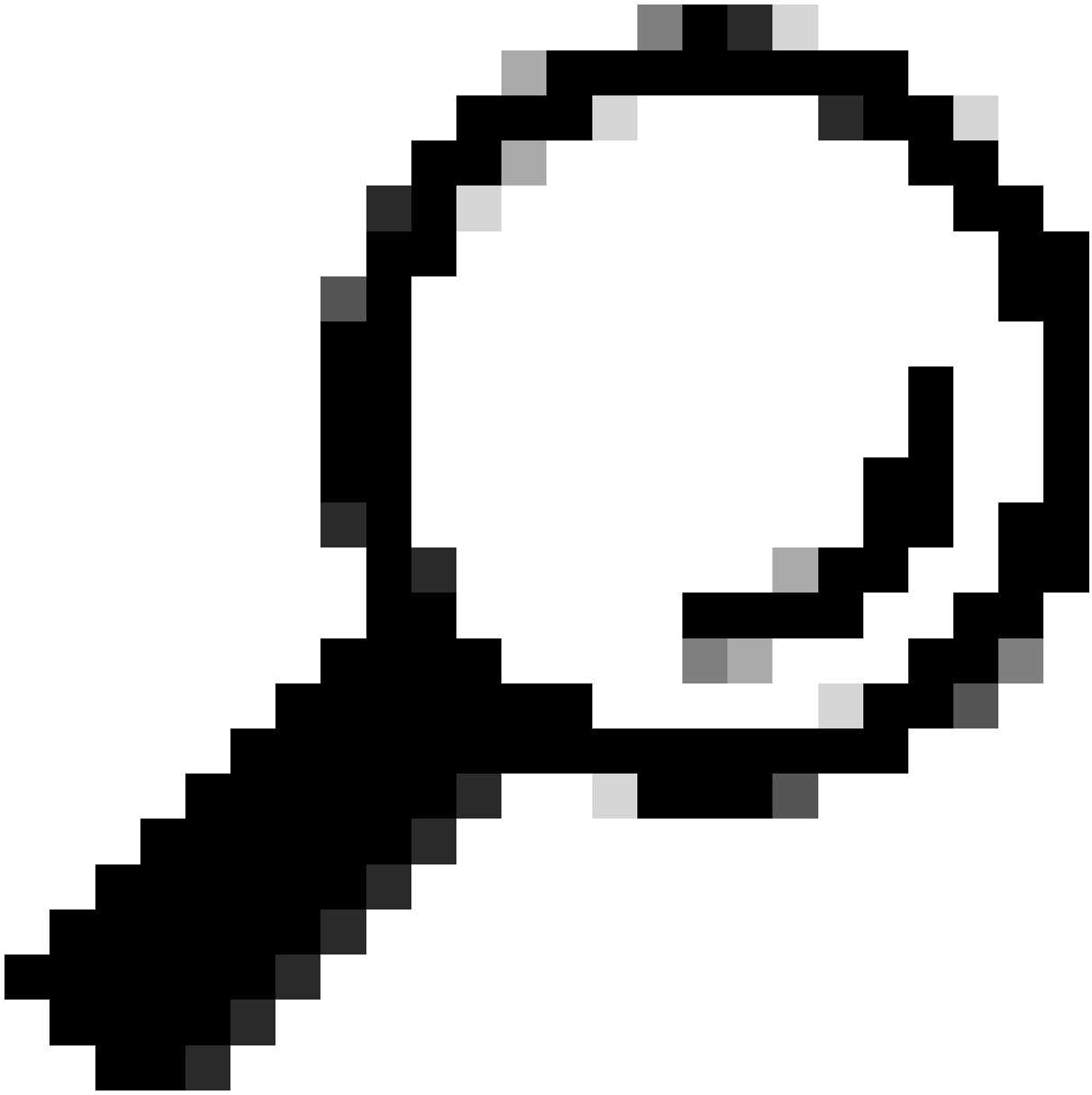
팁: 모든 디바이스 유형 및 모든 위치는 ISE에서 제공하는 기본 계층입니다. 사용자 자신의 계층을 추가 하고 정책 조건에서 나중에 사용 할 수 있는 네트워크 장치를 식별 하는 데 다양 한 구성 요소를 정의 할 수 있습니다

2단계. 이제 Cisco IOS XE 디바이스를 네트워크 디바이스로 추가합니다. Work Centers(작업 센터) > Device Administration(디바이스 관리) > Network Resources(네트워크 리소스) > Network Devices(네트워크 디바이스)로 이동합니다. 새 네트워크 디바이스를 추가하려면 Add(추가)를 클릭합니다. 이 테스트의 경우 SVS_BRPASR1K입니다.



3단계. 디바이스의 IP 주소를 입력하고 디바이스의 위치 및 디바이스 유형(IOS XE)을 매핑하십시오. 마지막으로, TACACS+ over TLS 인증 설정을 활성화합니다.



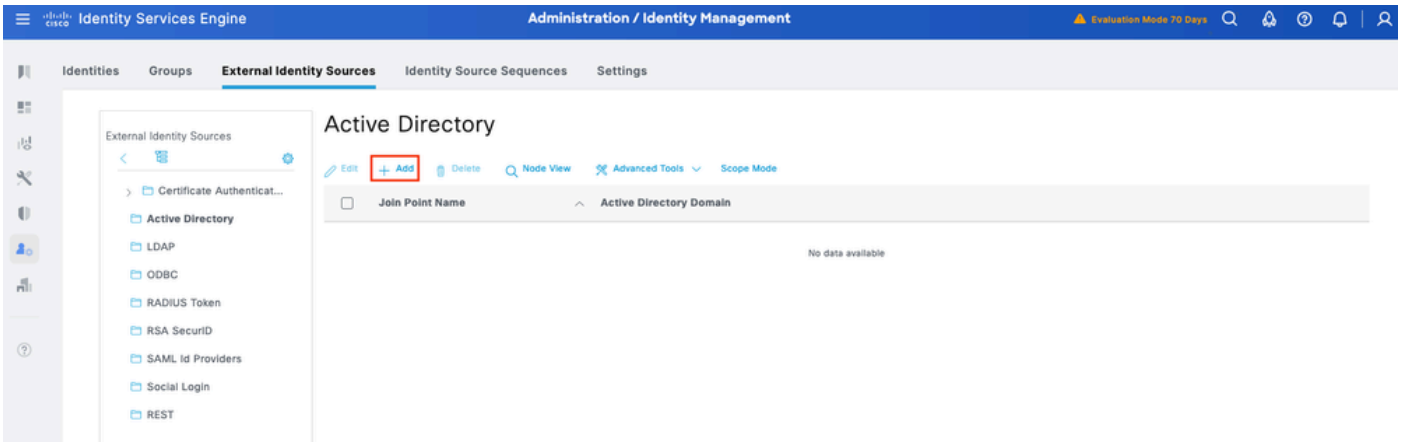


팁: 디바이스에 명령을 전송할 때마다 TCP 세션이 다시 시작되지 않도록 하려면 Enable Single Connect Mode(단일 연결 모드 활성화)를 사용하는 것이 좋습니다.

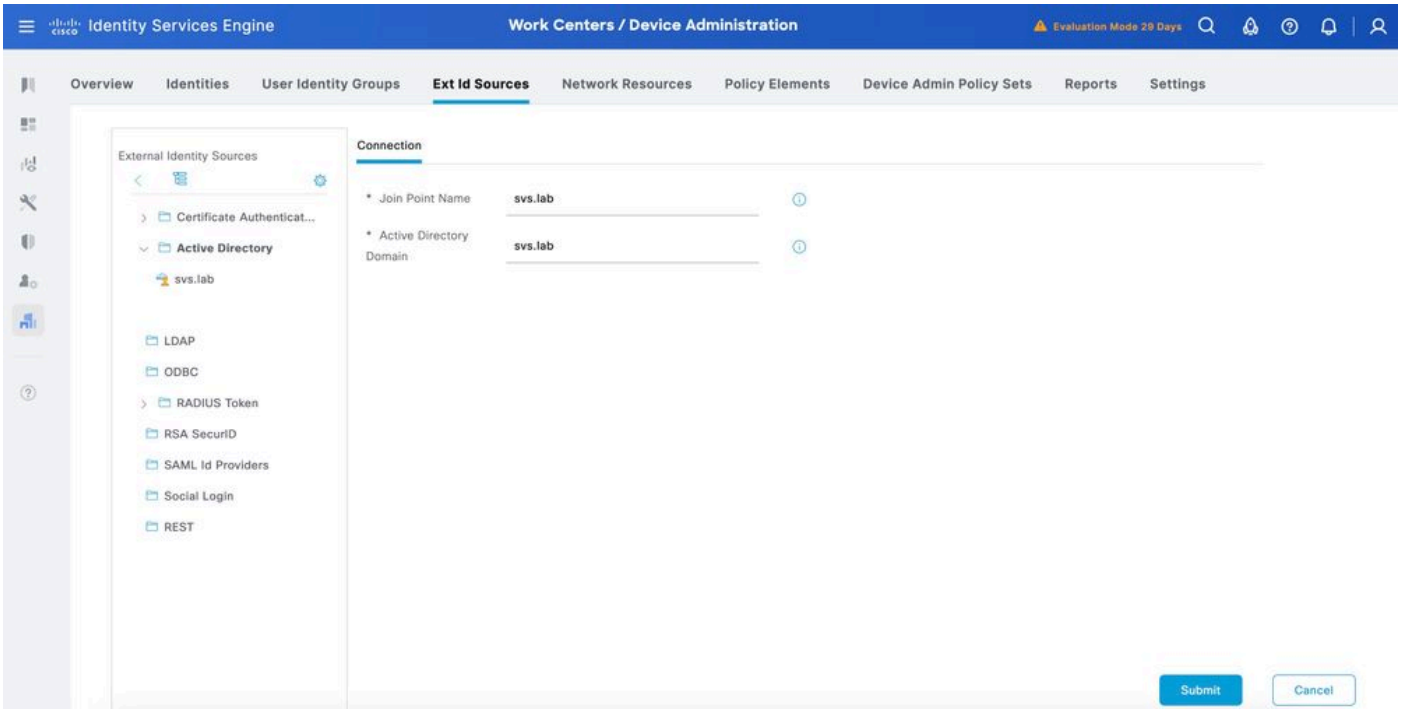
ID 저장소 구성

이 섹션에서는 디바이스 관리자를 위한 ID 저장소를 정의합니다. 이 저장소는 ISE 내부 사용자 및 지원되는 모든 외부 ID 소스가 될 수 있습니다. 여기서는 외부 ID 소스인 AD(Active Directory)를 사용합니다.

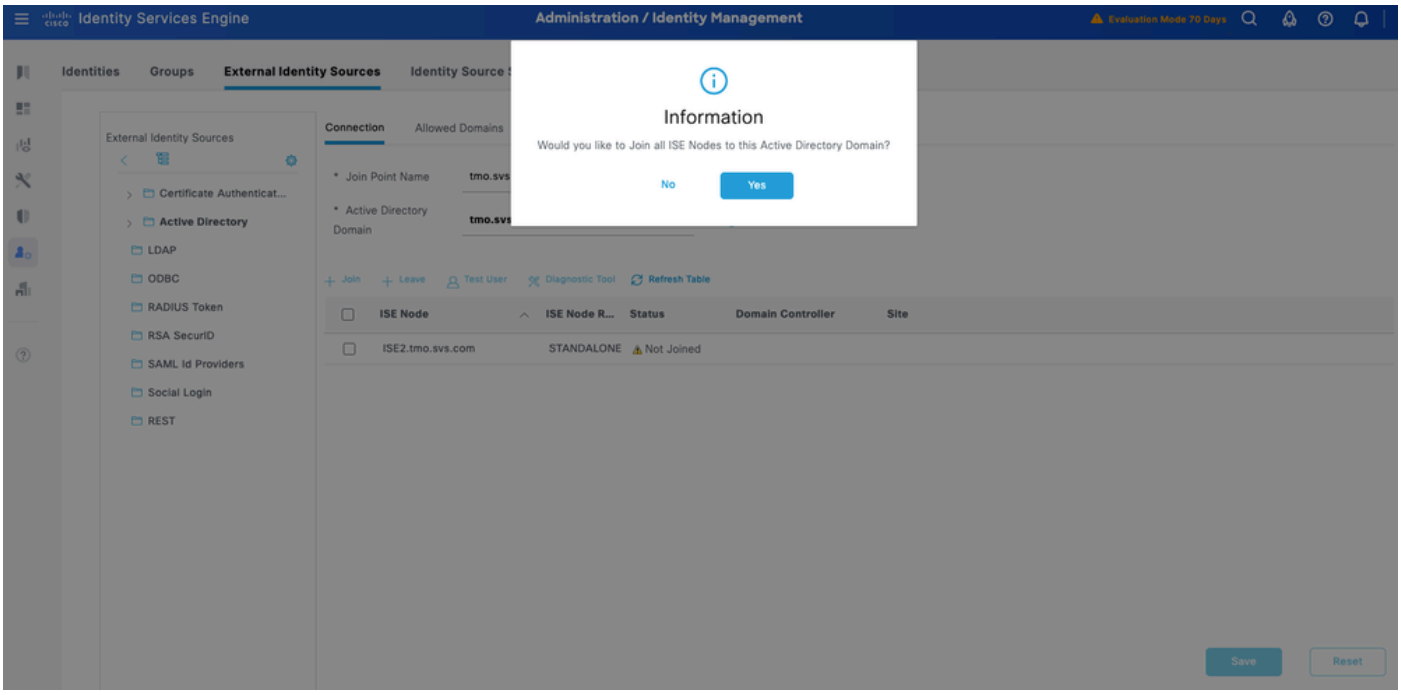
1단계. Administration(관리) > Identity Management(ID 관리) > External Identity Stores(외부 ID 저장소) > Active Directory로 이동합니다. 새 AD 조인트 포인트를 정의하려면 Add(추가)를 클릭합니다.



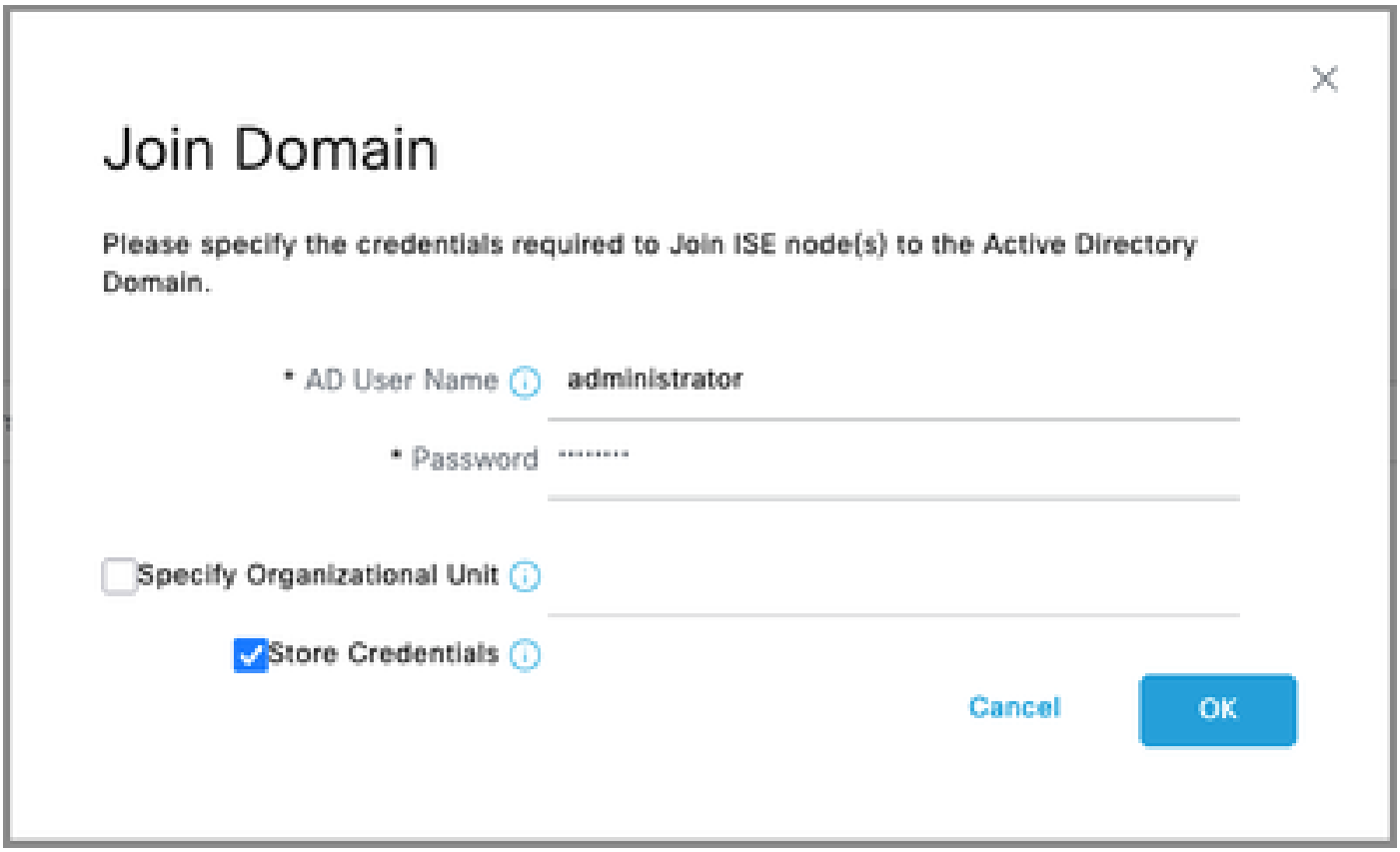
2단계. Join Point 이름과 AD 도메인 이름을 지정하고 Submit(제출)을 클릭합니다.

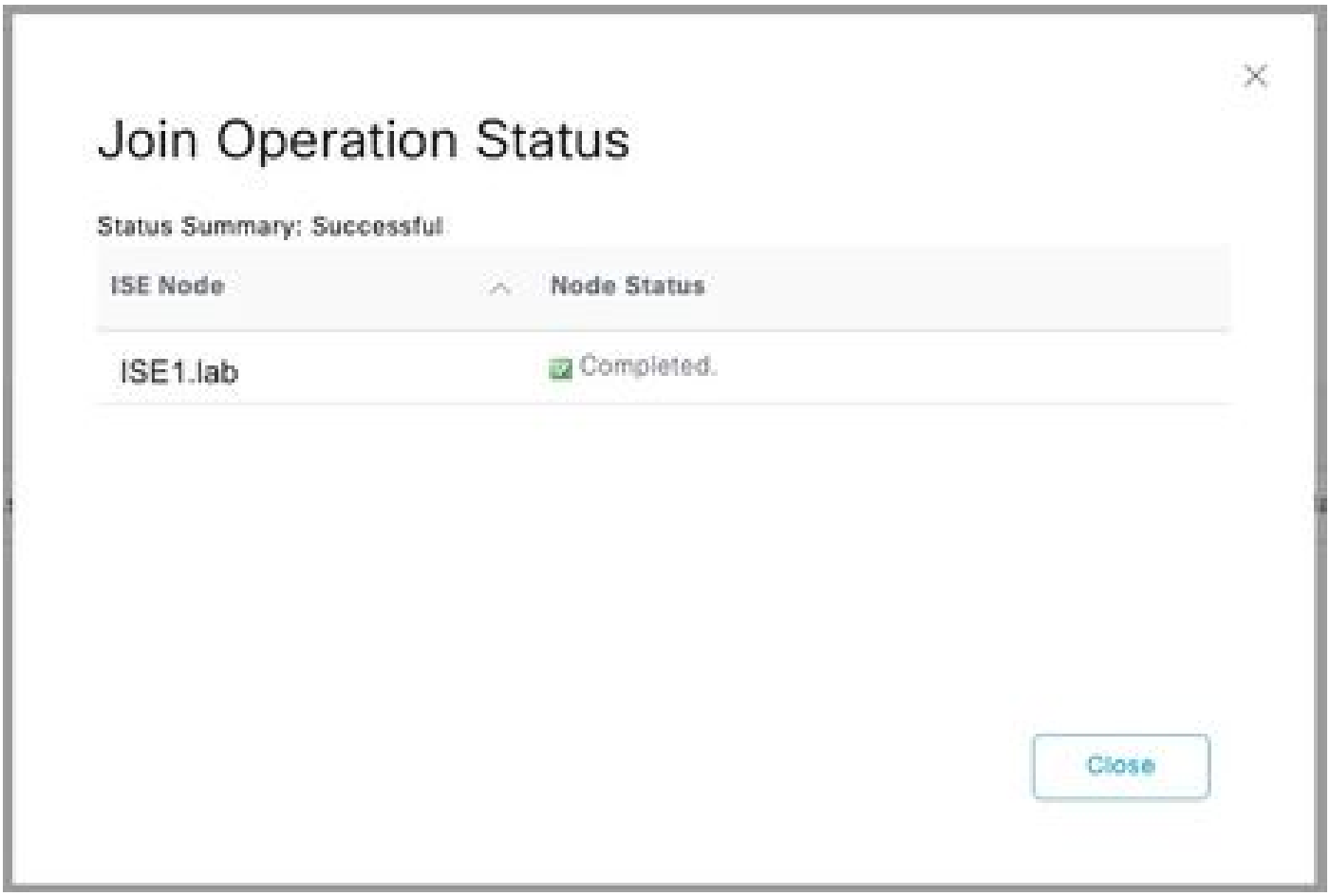


3단계. Would you like all ISE Nodes to this Active Directory Domain?(이 Active Directory 도메인에 모든 ISE 노드를 조인하시겠습니까?) 메시지가 나타나면 Yes(예)를 클릭합니다.

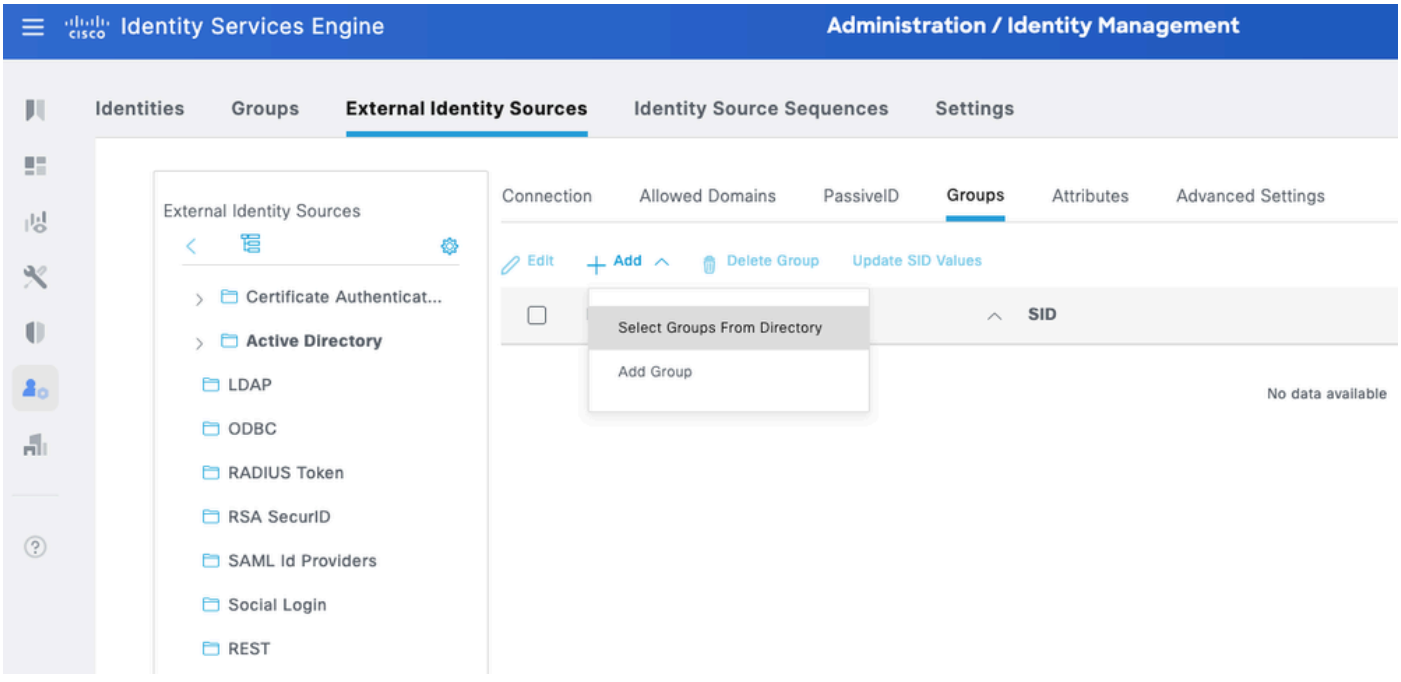


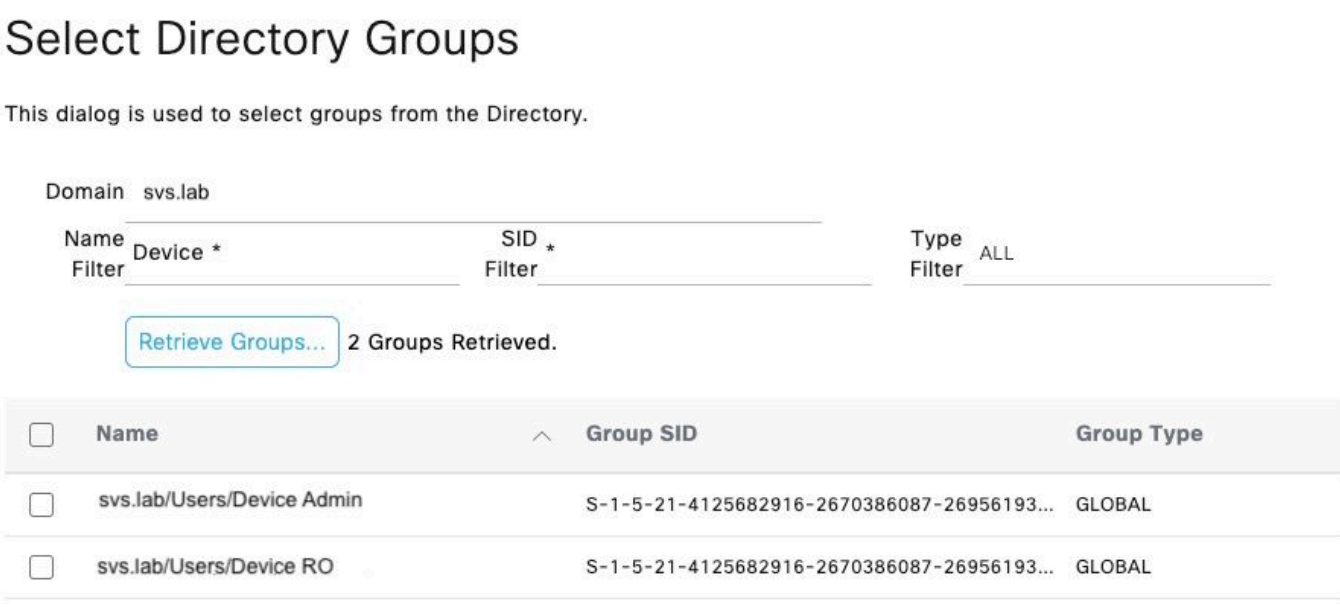
4단계. AD 조인 권한이 있는 자격 증명을 입력하고 ISE를 AD에 조인합니다. 작동 여부를 확인하려면 Status(상태)를 선택합니다.

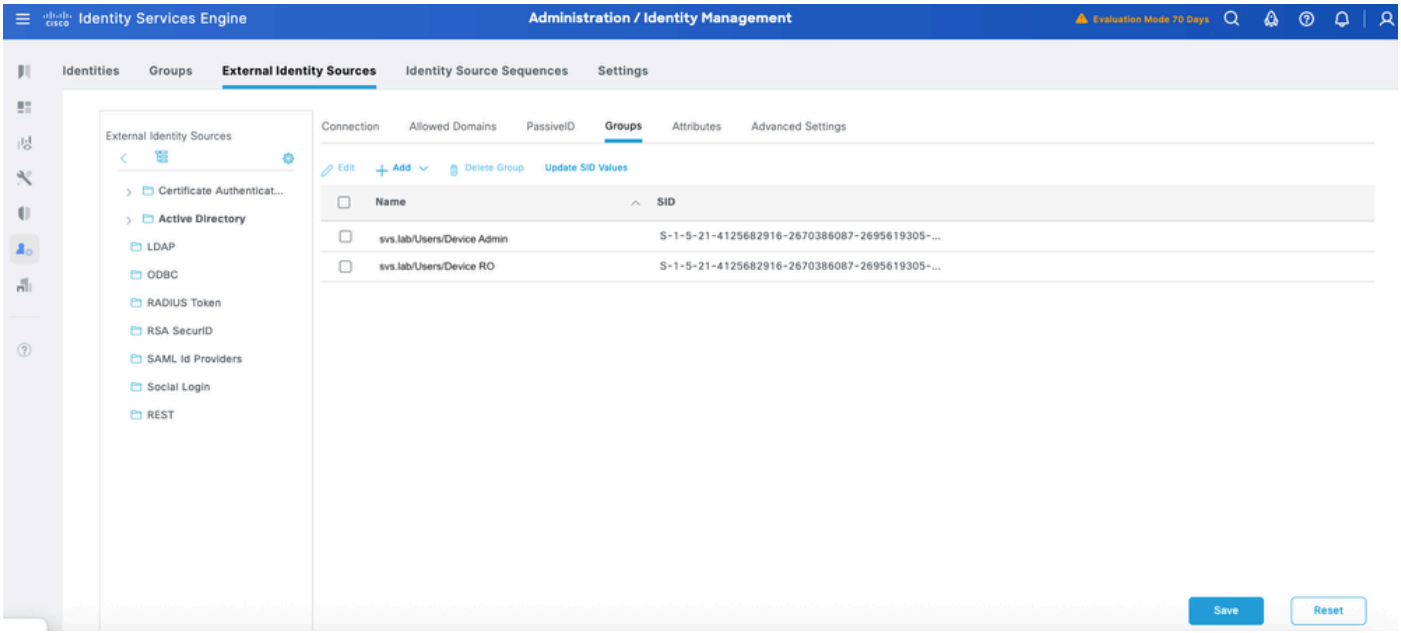




5단계. Groups(그룹) 탭으로 이동한 다음 Add(추가)를 클릭하여 디바이스 액세스 권한이 부여된 사용자에 따라 필요한 모든 그룹을 가져옵니다. 이 예는 이 가이드의 권한 부여 정책에 사용된 그룹을 보여줍니다







TACACS+ 프로파일 구성

TACACS+ 프로필을 Cisco IOS XE 디바이스의 두 가지 주요 사용자 역할에 매핑합니다.

- Root System Administrator(루트 시스템 관리자) - 디바이스에서 권한이 가장 높은 역할입니다. 루트 시스템 관리자 역할의 사용자는 모든 시스템 명령 및 컨피그레이션 기능에 대한 완전한 관리 액세스 권한을 갖습니다.
- 운영자 - 이 역할은 모니터링 및 문제 해결을 위해 시스템에 대한 읽기 전용 액세스가 필요한 사용자를 위한 것입니다.

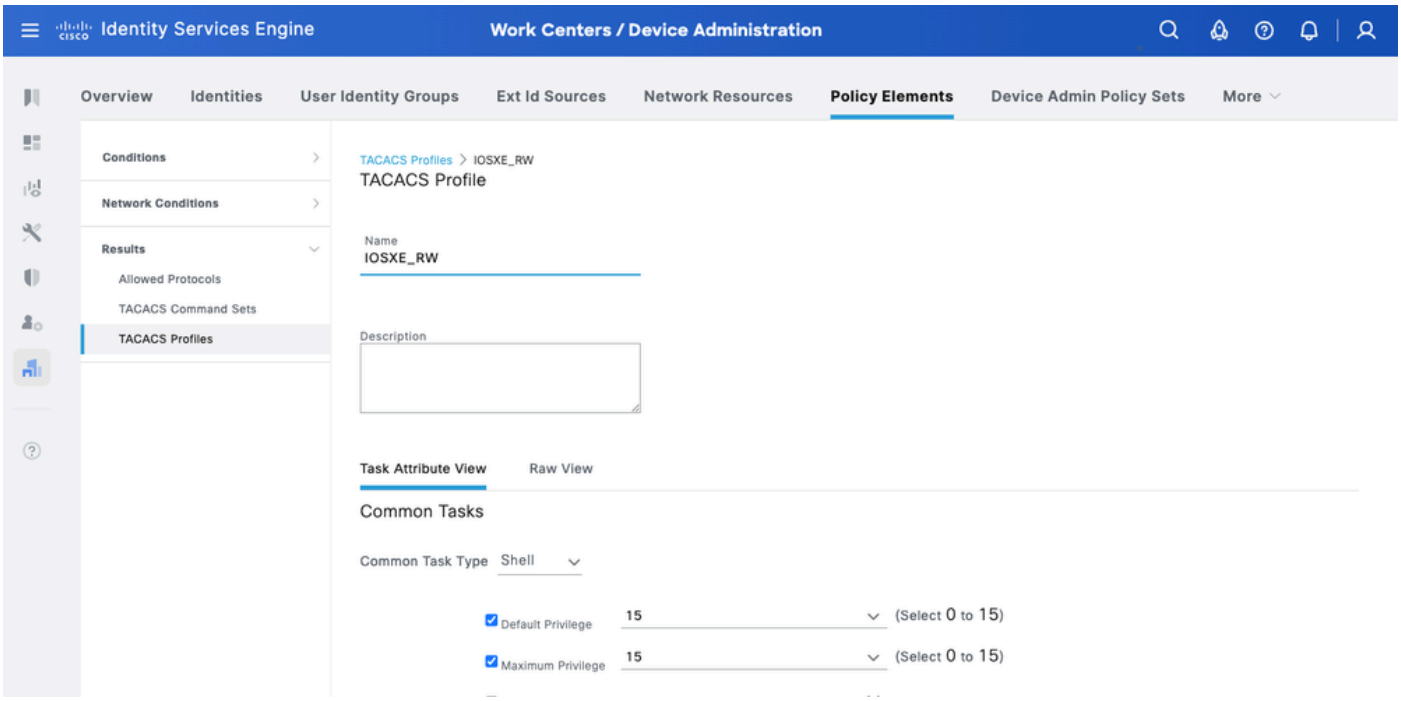
이는 2개의 TACACS+ 프로필로 정의됩니다. IOS XE_RW 및 IOSXR_RO.

IOS XE_RW - 관리자 프로필

1단계 Work Centers(작업 센터) > Device Administration(디바이스 관리) > Policy Elements(정책 요소) > Results(결과) > TACACS Profiles(TACACS 프로파일)로 이동합니다. 새 TACACS 프로파일을 추가하고 이름을 IOS XE_RW로 지정합니다.

2단계. Default Privilege(기본 권한) 및 Maximum Privilege(최대 권한)를 선택하여 15로 설정합니다.

3단계. 컨피그레이션을 확인하고 저장합니다.

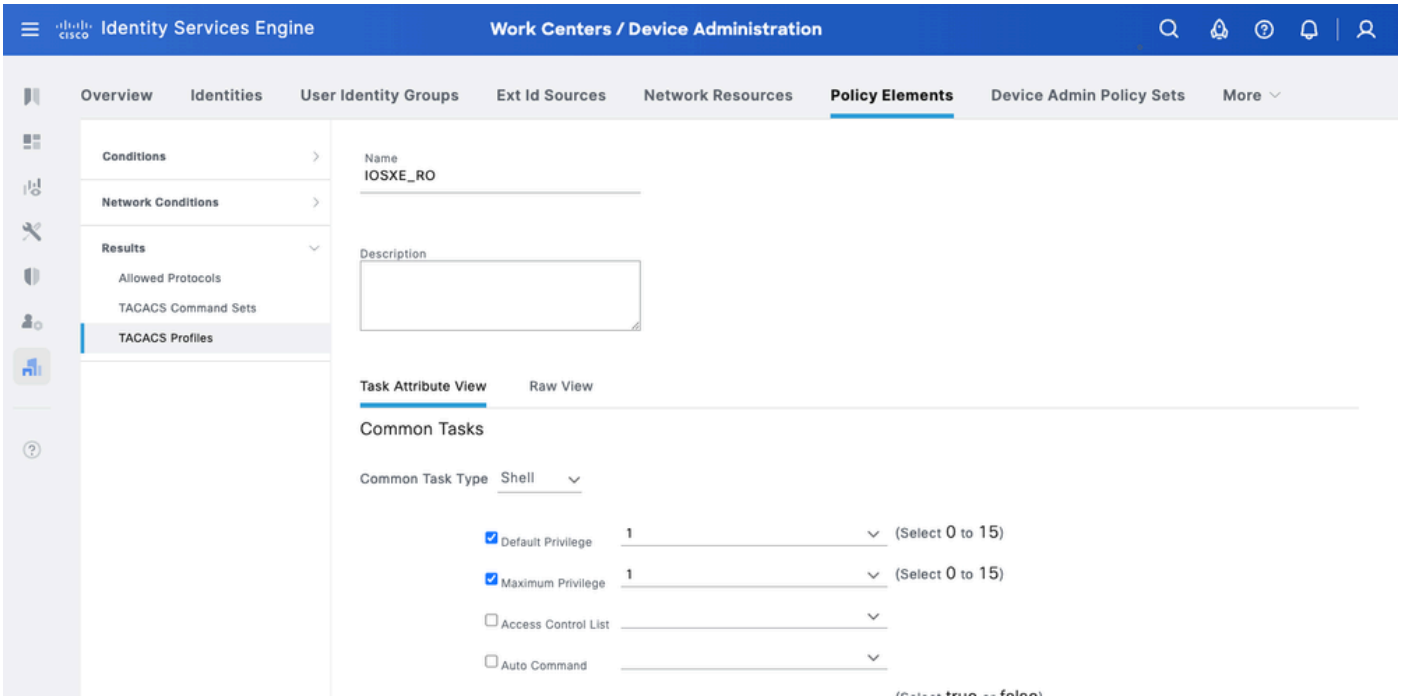


IOS XE_RO - 운영자 프로파일

1단계 Work Centers(작업 센터) > Device Administration(디바이스 관리) > Policy Elements(정책 요소) > Results(결과) > TACACS Profiles(TACACS 프로파일)로 이동합니다. 새 TACACS 프로파일을 추가하고 이름을 IOS XE_RO로 지정합니다.

2단계. Default Privilege(기본 권한) 및 Maximum Privilege(최대 권한)를 선택하여 1로 설정합니다.

3단계. 컨피그레이션을 확인하고 저장합니다.



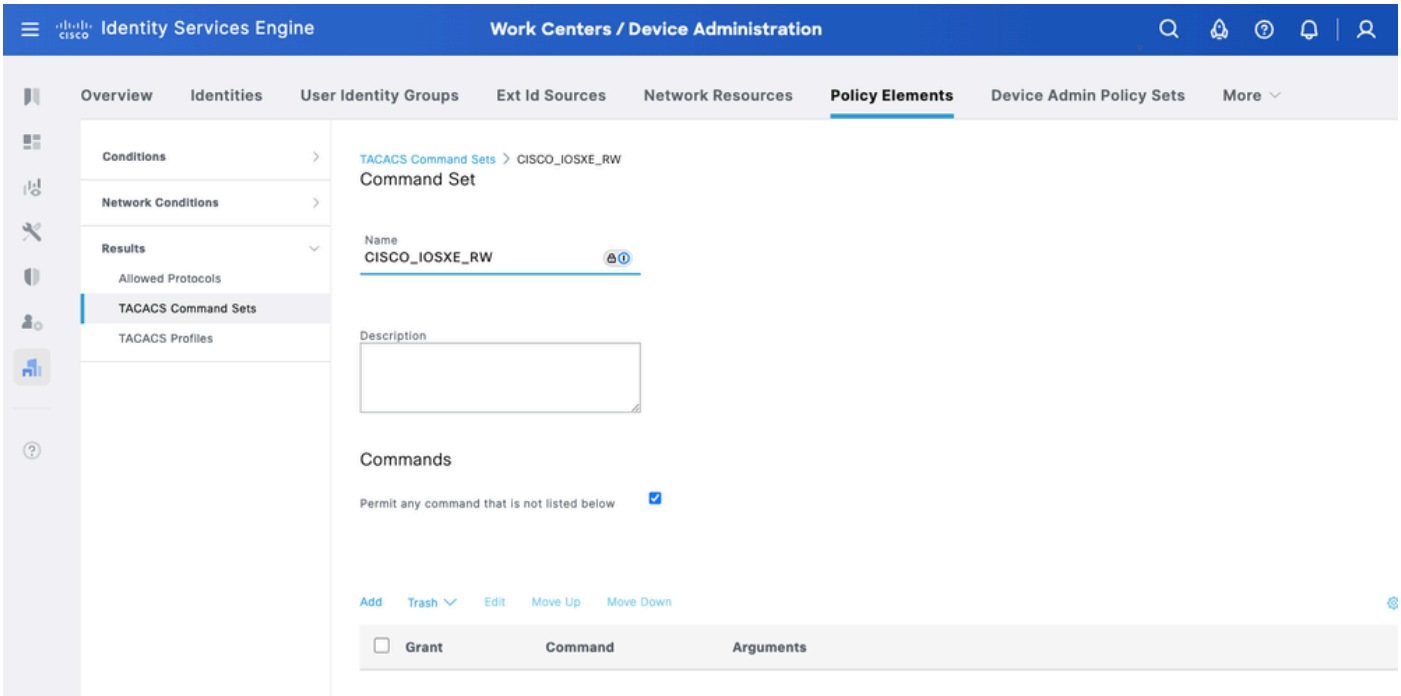
TACACS+ 명령 집합 구성

이는 2개의 TACACS+ 명령 세트로 정의됩니다. CISCO_IOS XE_RW 및 CISCO_IOS XE_RO.

CISCO_IOS XE_RW - 관리자 명령 집합

1단계. Work Centers(작업 센터) > Device Administration(디바이스 관리) > Policy Elements(정책 요소) > Results(결과) > TACACS Command Sets(TACACS 명령 집합)로 이동합니다. 새 TACACS 명령 집합을 추가하고 이름을 CISCO_IOS XE_RW로 지정합니다.

2단계. 아래에 나열되지 않은 모든 명령 허용 확인란을 선택하고(관리자 역할에 대한 모든 명령 허용) 저장을 클릭합니다.



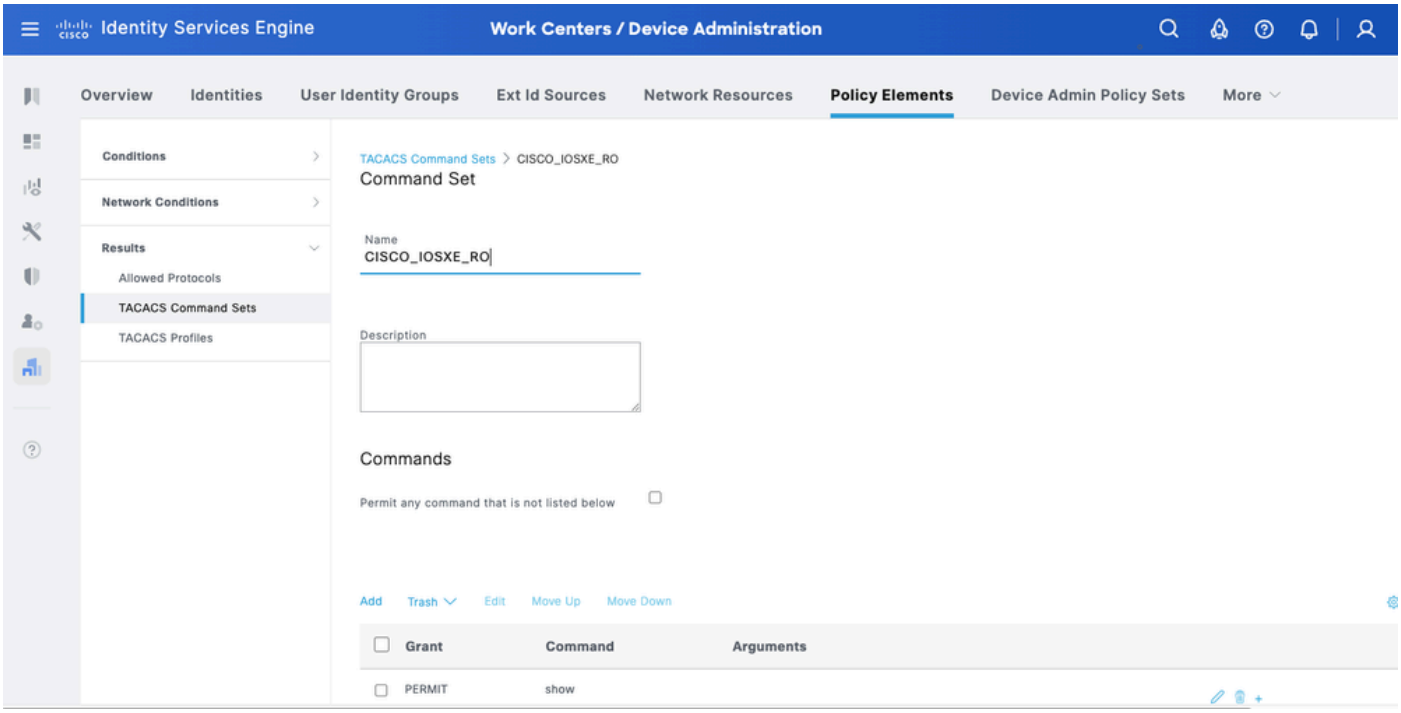
CISCO_IOS XE_RO - Operator 명령 집합

1단계 ISE UI에서 Work Centers(작업 센터) > Device Administration(디바이스 관리) > Policy Elements(정책 요소) > Results(결과) > TACACS Command Sets(TACACS 명령 집합)로 이동합니다. 새 TACACS 명령 집합을 추가하고 이름을 CISCO_IOS XE_RO로 지정합니다.

2단계. Commands(명령) 섹션에서 새 명령을 추가합니다.

3단계. Grant(권한 부여) 열의 드롭다운 목록에서 Permit(허용)을 선택하고 Command(명령) 열에 show를 입력합니다. 확인 화살표를 클릭합니다.

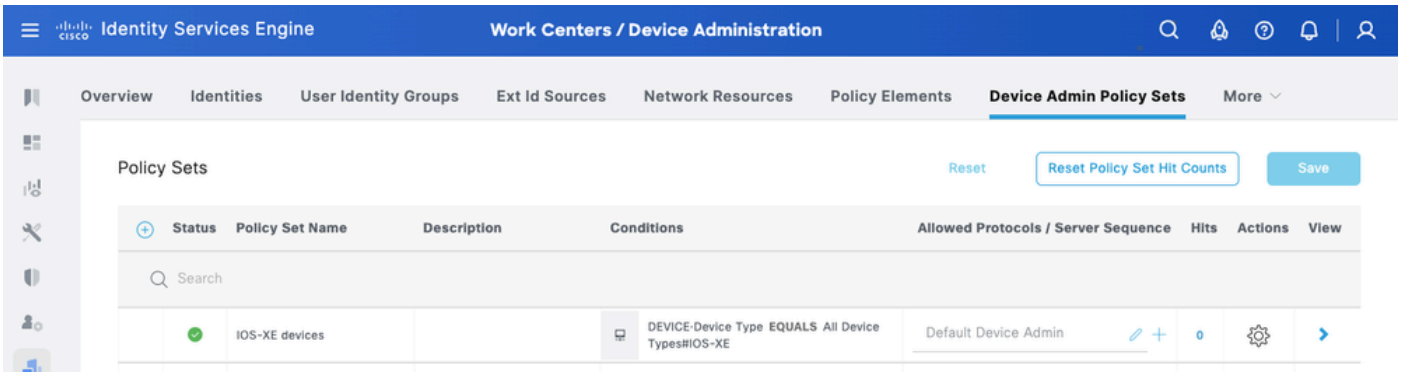
4단계. 데이터를 확인하고 저장을 누릅니다.



디바이스 관리 정책 집합 구성

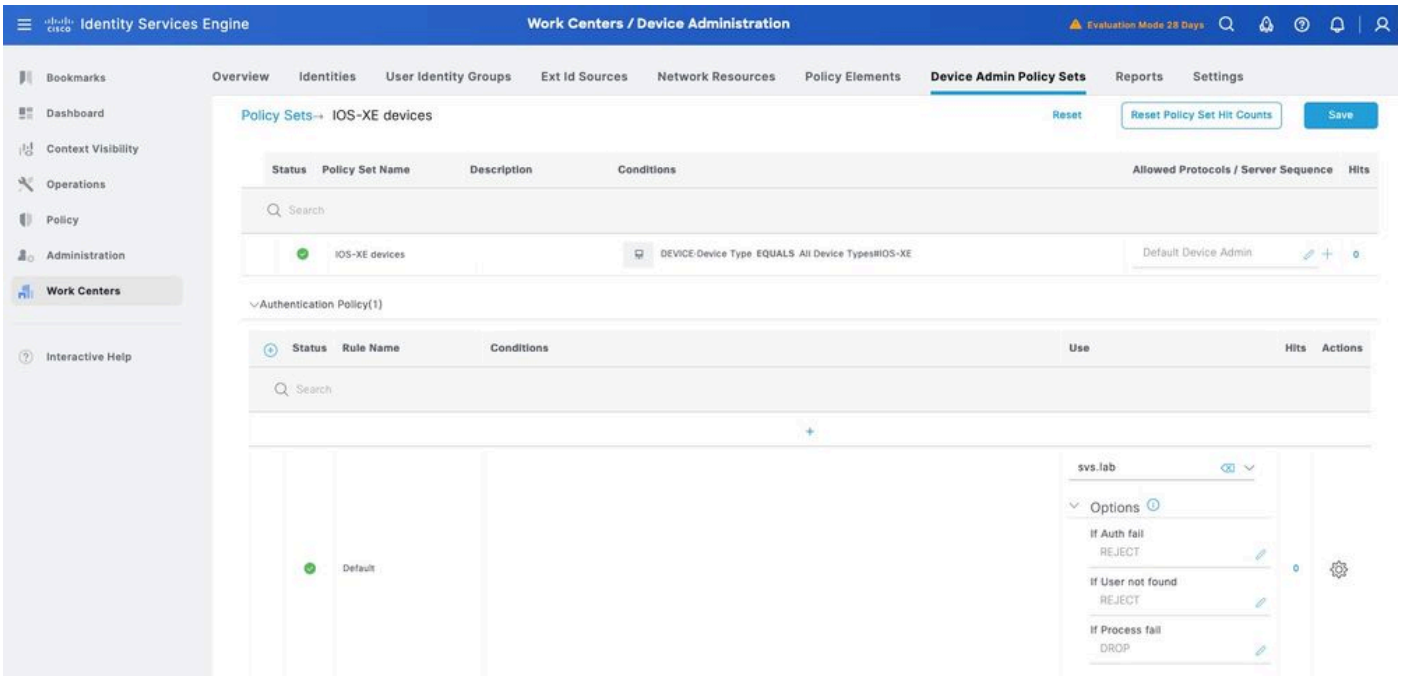
Policy Sets(정책 집합)는 Device Administration(디바이스 관리)에 대해 기본적으로 활성화되어 있습니다. 정책 집합은 TACACS 프로파일을 쉽게 적용할 수 있도록 디바이스 유형에 따라 정책을 구분할 수 있습니다.

1단계. Work Centers(작업 센터) > Device Administration(디바이스 관리) > Device Admin Policy Sets(디바이스 관리 정책 집합)로 이동합니다. 새 Policy Set IOS XE Devices를 추가합니다. 조건에서 DEVICE:Device Type EQUALS All Device Types#IOS XE를 지정합니다. Allowed Protocols(허용된 프로토콜) 아래에서 Default Device Admin(기본 디바이스 관리)을 선택합니다.



2단계. Save(저장)를 클릭하고 오른쪽 화살표를 클릭하여 이 정책 집합을 구성합니다.

3단계. 인증 정책을 생성합니다. 인증에서는 AD를 ID 저장소로 사용합니다. If Auth fail(인증 실패 경우), If User not found(사용자를 찾을 수 없는 경우) 및 If Process fail(프로세스 실패)에 기본 옵션을 그대로 둡니다.



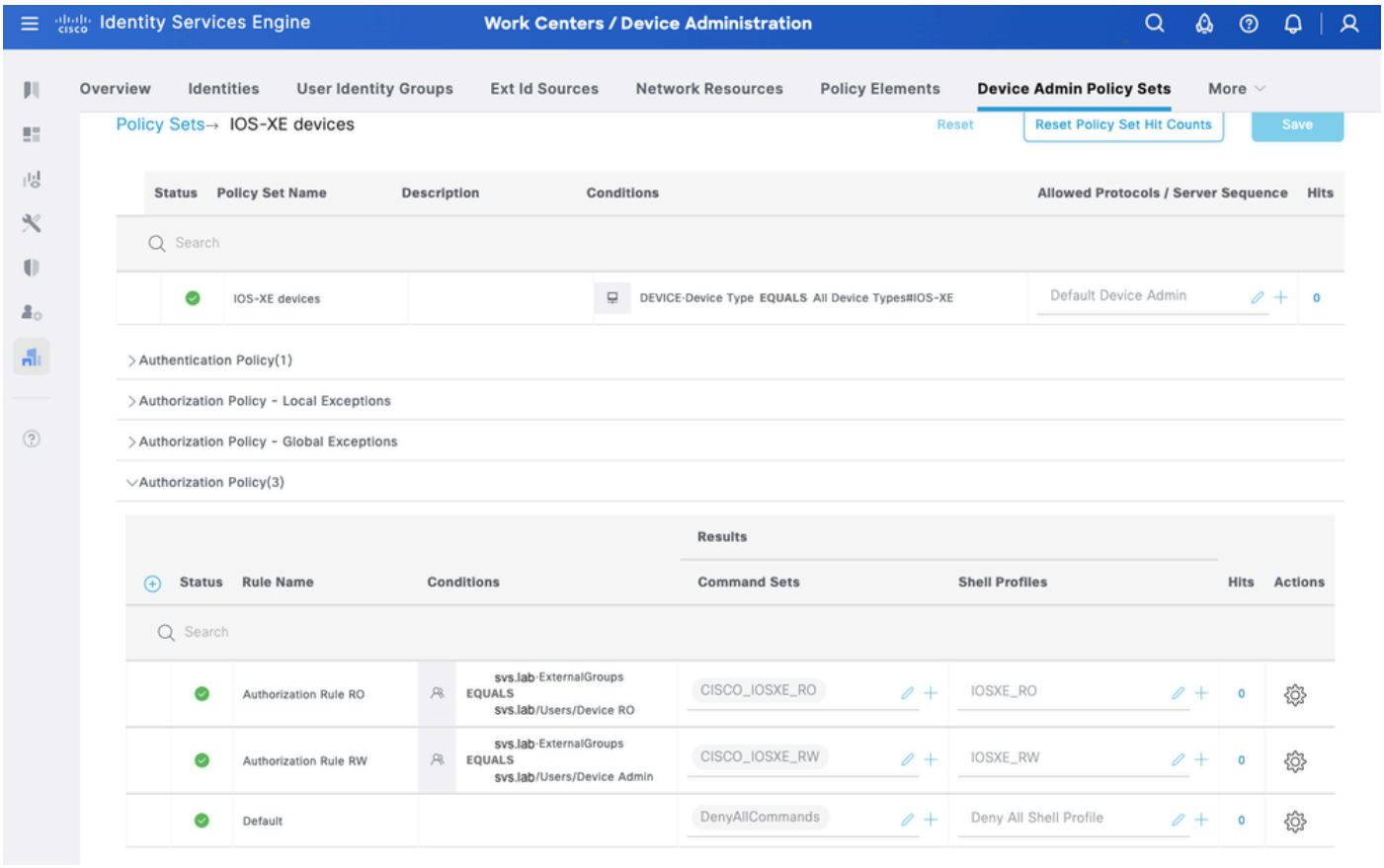
4단계. 권한 부여 정책을 정의합니다.

Active Directory(AD)의 사용자 그룹을 기반으로 권한 부여 정책을 생성합니다.

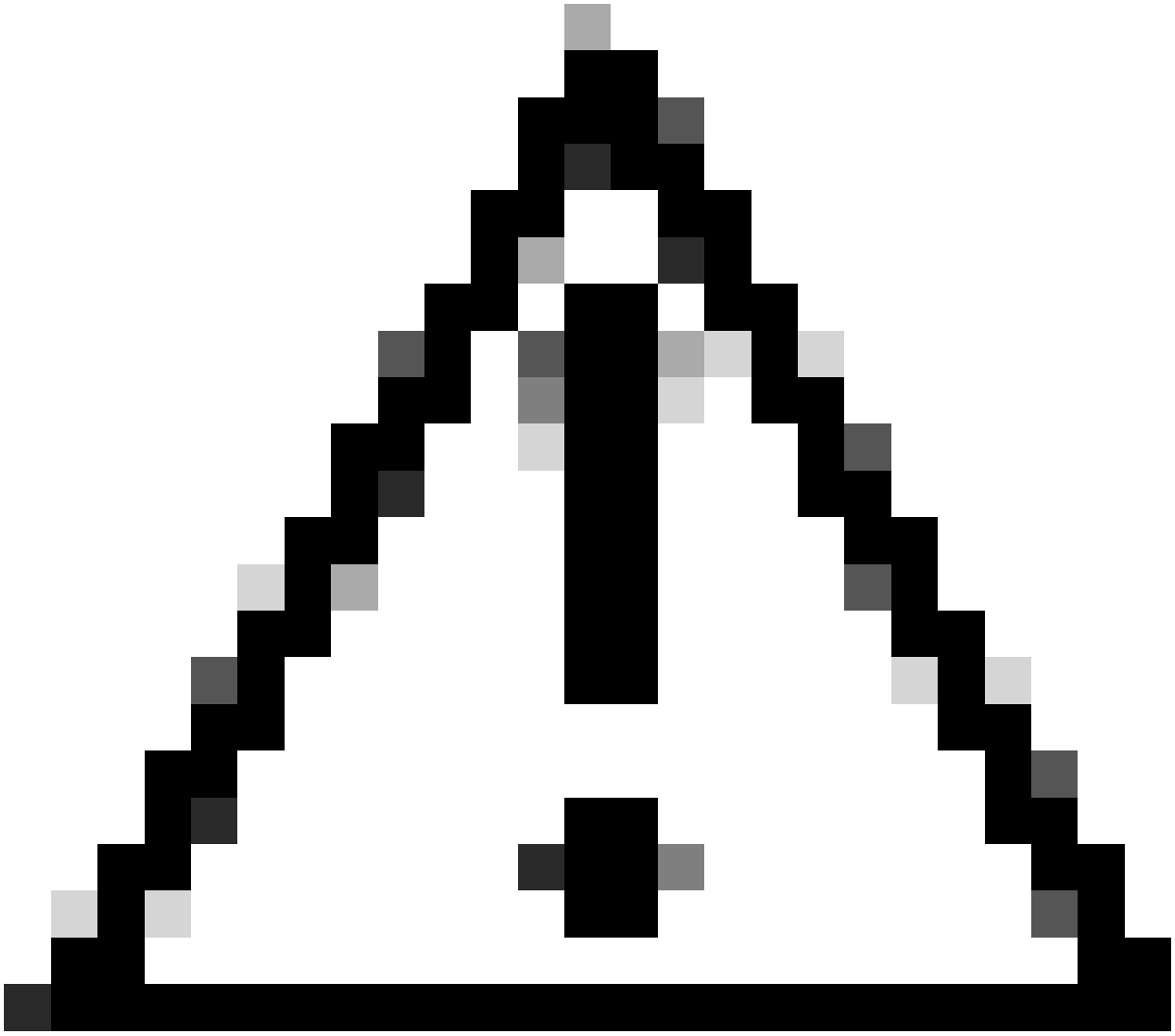
예를 들면 다음과 같습니다.

· AD 그룹 디바이스 RO의 사용자는 CISCO_IOSXR_RO 명령 집합 및 IOSXR_RO 셸 프로필이 할당됩니다.

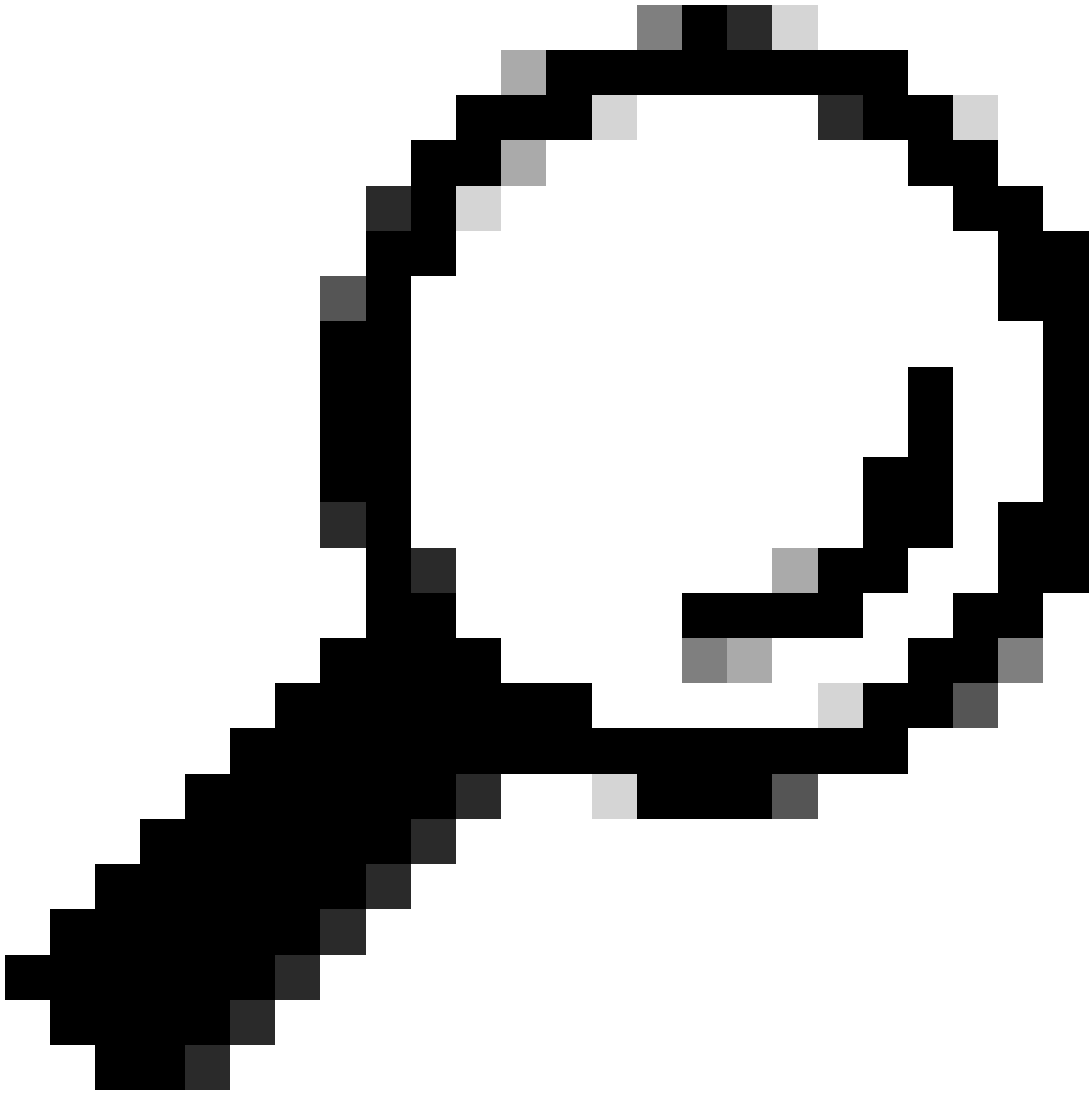
· AD 그룹 장치 관리의 사용자는 CISCO_IOSXR_RW 명령 집합 및 IOSXR_RW 셸 프로파일이 할당됩니다.



2부 - TLS 1.3을 통한 TACACS+용 Cisco IOS XE 구성



주의: 콘솔 연결에 연결할 수 있고 제대로 작동하는지 확인합니다.



팁: 디바이스에서 잠기지 않도록 컨피그레이션을 변경하는 동안 TACACS 대신 로컬 자격 증명을 사용하도록 임시 사용자를 구성하고 AAA 인증 및 권한 부여 방법을 변경하는 것이 좋습니다.

컨피그레이션 방법 1 - 디바이스 생성 키 쌍

TACACS+ 서버 컨피그레이션

1단계 도메인 이름을 구성하고 라우터 신뢰 지점에 사용되는 키 쌍을 생성합니다.

```
ip domain name svs.lab

crypto key generate ec keysize 256 label svs-256ec-key
```

신뢰 지점 컨피그레이션

1단계 라우터 신뢰 지점을 생성하고 키 쌍을 연결합니다.

```
crypto pki trustpoint svs_cat9k
  enrollment terminal pem
  subject-name C=US,ST=NC,L=RTP,O=Cisco,OU=SVS,CN=cat9k.svs.lab
  serial-number none
  ip-address none
  revocation-check none
  eckeypair svs-256ec-key
```

2단계. CA 인증서를 설치하여 신뢰 지점을 인증합니다.

<#root>

cat9k(config)#

crypto pki authenticate svs_cat9k

Enter the base 64 encoded CA certificate.

End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----

MIIFlDCCA3ygAwIBAgIIIM1OAsTaN/UwDQYJKoZIhvcNAQELBQAwajELMAkGA1UE
BhMCVVMxFzAVBgNVBAgTDk5vcnRoIENhcm9saW5hMRAwDgYDVQQHEwdSYWxlaWdo
MQ4wDAYDVQQKEwVDaXNjbzEMMAoGA1UECxMDU1ZTMRIwEAYDVQQDEwlTVlMgTGFi
QOEwHhcNMjUwNDI4MTcwNTAwWhcNMzUwNDI4MTcwNTAwWjBqMQswCQYDVQQGEwJV
UzEXMBUGA1UECBMOTm9ydGggQ2Fyb2xpbmExEDAOBgNVBAcTB1JhbGVpZ2gxDjAM
BgNVBAoTBUNpc2NvMQwwCgYDVQQLEwNTVlMxEjAQBgNVBAMTCVNWUyBMYWJDQTCC
AiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBAJvZUOyn2vIn6gKbx3M7vaRq
2YjwZlzSH6EkEvxnJTy+kksiFD33GyHQepk7vfp4NFU5OtQ4HC7t/AOv9grDa3QW
VwvV4MBbJhFM3sOJ/ejgDYcMZhIAaPyOZo5WLboOkXEiKjPLatkXojB8FVrhLF3O
jMBSqwa4/Wlniy5S+7s4FFxsCf2OCOWfBAsnrs0tatIIhmcnx+VLJP7MRm8fOw4m
mutNo7IhbJSrgAFXmjlbBjMmgspObULo/wxMHdTbtPBf11HRHTkNIo3qyO4UADL2
WpoGhgT/FaxxBo2UBcnYVaP+jjRE0NYT973MCbVAAxtNVU6bEBROz+LWniACzupm
+qh23SL43uW5A3iSw/BuU1E9p7BOe8oDNKU6gXlojKyLP/gC7j8AePO3ir+KZui8
b8X4iYn/67SbzZFhwxn3chkW4JYhQ4AImW1An2Q1+DMoZL7zRtSqQ3g9ZqRIMzQN
gJ+kQXe7QtT/u6m1MrtjE3gAEVpL334rTIxy9hpKZIkB86t2ZA3JX8CLsbCa13sA
z1XCoONX+6a1ekmXuAOI+t3c1sNbN2AtFi4cJovTAO1xh6OI4QnK+MNQKpTjt/E4
ydHlOrrurXsZummj9QBnkX4pqY7cDLHhdMKpbjDwg7jVLl783nTc9wYptQEPi5sw

```
83g9EMgKV0ARIiVUa/qlAgMBAAGjPjA8MAwGA1UdEwQFMAMBAf8wEQYJYIZIAYb4
QgEBBAQDAgAHMBkGCWCGSAGG+EIBDQQMFgpTVlMgTGFiIENBMA0GCSqGSIb3DQEB
CwUAA4ICAQAIT3O8oL2L6j/7Kk9VdcouuaBsN9o2pNEk3KXeZ8ykarNoxa87sFYr
AwXIwfAtk8uEHfnWu1QcZ3LkEJM9rHVCZuKsYd3D6qojo54HTpxRLgo5oKOdGayi
iSEkSSX9qyfLfINHR2JSVqJU6jLsy86X7q7RmIPMS7XfHzuddFNI4YDoXRX67X+v
O+ja6zTQqjO6lqJhmrSkyFbYf/ZTpe4d1OzJsZjNsNOr8bF9nOA/7qNZLp3Z3cpU
PUOKdbiSvRqnPw3e8TfITVmAzcx8COI2SrYFMSUazolVBvDy+xRKxyAtMbneGz6n
YdykCimThCKoKwp/pWpYBEqIEOf5ay1PKURO/8aj/B7aluJapXkmnj5qPeGhNOpB
Q9r14reov4so2EspkXS7CrH9yGfpIyTprokz1UvZBZ8vloI7YZmjFmem+5rT6Gnk
eU/lX7nV61SYG5W5K+I8uaKuyBHOMn7Amy3DYL5c5GJBqxpSZERbLXV+Q1tIgRU8
8ggzlPOdsS/i6Lo7ypYXOeB9HgVDCkzQsLXQuHGj/2WsgPgdRcjkvnyURk4Jx+Ib
xDrmo7eOXPpSW4l72a6Kl8CR3U2Cr4wsuvndPEq/qd2NRSBWffF0XE/AJHQG7STT
HaXLU9r2Ko6O3oecu8ysGTwLlIt/9T1/FOb0xZRugWcpJrVoTgDGuA==

-----END CERTIFICATE-----

Certificate has the following attributes:
   Fingerprint MD5: D9C404B2 EC08A260 EC3539E7 F54ED17D
   Fingerprint SHA1: 0EB181E9 5A3ED780 3BC5A805 9A854A95 C83AC737

% Do you accept this certificate? [yes/no]:

yes


Trustpoint CA certificate accepted.
% Certificate successfully imported

cat9k(config)#
```

3단계. CSR(Certificate Signing Request)을 생성합니다.

<#root>

```
cat9k(config)#

crypto pki enroll svs_cat9k


% Start certificate enrollment ..

% The subject name in the certificate will include: C=US,ST=NC,L=RTP,O=Cisco,OU=SVS,CN=cat9k.svs.lab
% The subject name in the certificate will include: cat9k.svs.lab
Display Certificate Request to terminal? [yes/no]:

yes


Certificate Request follows:

-----BEGIN CERTIFICATE REQUEST-----
MIIBfDCCASMCAQAwgYQxGjAYBgNVBAMTEWNhdDlrLnRtby5zdnMuY29tMQwwCgYD
VQQLEwNTVlMxDjAMBgNVBAoTBUNpc2NvMQwwCgYDVQQHEwNSVFAxCzAJBgNVBAgT
Ak5DMQswCQYDVQQGEwJVUzEgMB4GCSqGSIb3DQEJAhYRY2F0OWsudG1vLnN2cy5j
b20wWTATBgcqhkjOPQIBBggqhkjOPQMBBwNCAATpYE7atscrtl4ddevCh3UgxjYi
4N4oBGWrpJBctKy4so8V5i6RXDt7kHgPzp14Qnf20bcXVODE1wtTAHHBrIXqoDww
OgYJKoZIhvcNAQkOMS0wKzAcBgNVHREEFTATghFjYXQ5ay50bW8uc3ZzLmNvbTAL
BgNVHQ8EBAMCB4AwCgYIKoZIzj0EAwQDRwAwRAIgZqP2QTwM3ZZrmIphJ7+jSTER
4OkTx2DiVs1c1Xf+vR4CIBcSb18DIYz84DmgMHUaf778/cmpe9cWakvdaxMWseBH
-----END CERTIFICATE REQUEST-----
```

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

Redisplay enrollment request? [yes/no]:
```

no

```
cat9k(config)#
```

4단계. CA 서명 인증서를 가져옵니다.

<#root>

```
cat9k(config)#
```

crypto pki import svs_cat9k certificate

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

-----BEGIN CERTIFICATE-----
MIID8zCCAdugAwIBAgIIKfdYWg5WpskwDQYJKoZIhvcNAQELBQAwajELMAkGA1UE
BhMCVVMxFzAVBgNVBAgTDk5vcnRoIENhcm9saW5hMRAwDgYDVQQHEwdSYWxlaWdo
MQ4wDAYDVQQKEwVDaXNjbzEMMAoGA1UECxMDU1ZTMRIwEAYDVQQDEwlTVlMgTGFi
QOEwHhcNMjUwNTEOMTUxMjAwWhcNMjYwNTEOMTUxMjAwWjCBhDEaMBgGA1UEAxMR
Y2FOOWsudG1vLnN2cy5jb2OxDDAKBgNVBAsTA1NWUzEOMAwGA1UEChMFQ2lzY28x
DDAKBgNVBAcTA1JUUDELMAkGA1UECBMCTkMxCzAJBgNVBAYTAlVTMSAwHgYJKoZI
hvcNAQkCFhFjYXQ5ay5Obw8uc3ZzLmNvbTBZMBMGByqGSM49AgEGCCqGSM49AwEH
AOIABOlgTtq2xyu2Xh1168KHdSDGNiLg3igEZaukkFyOrLiyjxXmLpFcO3uQeA/O
nXhCd/bRtxdU4MTXC1MAccGsheqjTTBLMB4GCWCGSAGG+EIBDQQRFg94Y2EgY2Vy
dGlmaWNhdGUwHAYDVR0RBBUwE4IRY2FOOWsudG1vLnN2cy5jb2OwCwYDVR0PBAQD
AgeAMA0GCSqGSIb3DQEBCwUAA4ICAQBObgKVykeyVC9UsvuuOAUsGaZHGwy2H9Yd
m5vIaui6PJczkCzIoAIghHPGQhIgpEcRqtGyXPZ2r8TCJP11WXNN/G73sFyWAhzY
RtmIM5KIojiDHLtifPayxv9juDuOZRx+wYR2PIQ5eLv1bafg7K8E82sqOCfOtcPr
OcONU8UCxqObdOgu4XsdBN1+wcWFqeQSDLmP7nxvhOOm/LXwCWUHwgVioOAuU2Fe
k5NthtvdxNAhRAImQdTyq6u/yB7vwTwJHcRiJc5USsyzCsTBb6RvL+HsXqBgXGc5
lxCSoLtYOdUxFIpJyK2MOZBY2zq2cNSc8Xbso5/OEQmnHtpWPvij4rSPUhQSY+4m
Qq2Sn3iqf4mGh/AO8T4iXfWDWfNezh7ZxMsCSCK/ZRlELZ2hj6OfzwX1H27Uf8XU
ecrOWx+WzRn7LVRCaGQzFkukfi8S4DLLNtxnNHfsLBVX5yHXCLEL+CQ7n8Z/pxcB
VVrPitwN3ZbO9poZyWiRLTnBsb42xNaWiL9bjQznAOiTDfmfFFourBsaAioz7ouY
2r1Mh+OpE83Uu+4lOTMawDgGiEv7iaiJ6xWc95EC+AdmOx3FvBXMtIM9qr7WwHW6
3C2hVYHJH254elV5+H8iiz7rovEPm8ZDsnvYpJn4Km3iDvBNqp/vvAHOFcyXrvG6
3i/1b9erGQ==
-----END CERTIFICATE-----

% Router Certificate successfully imported

cat9k(config)#
```

TACACS 및 AAA(TLS 컨피그레이션 포함)

1단계. TACACSS 서버 및 AAA 그룹을 생성하고 클라이언트(라우터) 신뢰 지점을 연결합니다.

```
tacacs server svs_tacacs
 address ipv4 10.225.253.209
 single-connection
 tls port 6049
 tls idle-timeout 60
 tls connection-timeout 60
 tls trustpoint client svs_cat9k
 tls ip tacacs source-interface GigabitEthernet0/0
 tls ip vrf forwarding Mgmt-vrf
!
aaa group server tacacs+ svs_tls
 server name svs_tacacs
 ip vrf forwarding Mgmt-vrf
!
tacacs-server directed-request
```

2단계. AAA 방법을 구성합니다.

```
aaa authentication login default group svs_tls local enable
aaa authentication login console local enable
aaa authentication enable default group svs_tls enable
aaa authorization config-commands
aaa authorization exec default group svs_tls local if-authenticated
aaa authorization commands 1 default group svs_tls local if-authenticated
aaa authorization commands 15 default group svs_tls
aaa accounting exec default start-stop group svs_tls
aaa accounting commands 1 default start-stop group svs_tls
aaa accounting commands 15 default start-stop group svs_tls
aaa session-id common
```

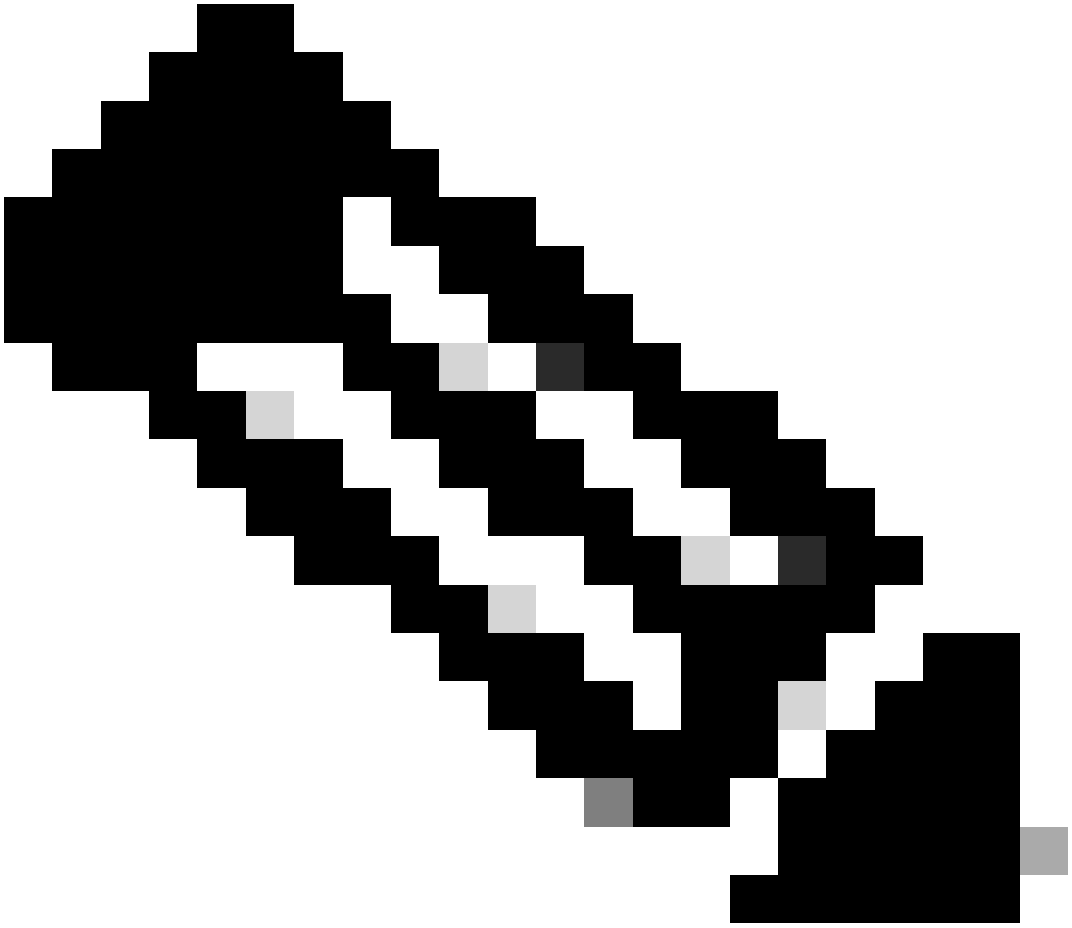
컨피그레이션 방법 2 - CA 생성 키 쌍

키와 디바이스 및 CA 인증서를 CSR 메서드 대신 PKCS#12 형식으로 직접 가져오는 경우 이 메서드를 사용할 수 있습니다.

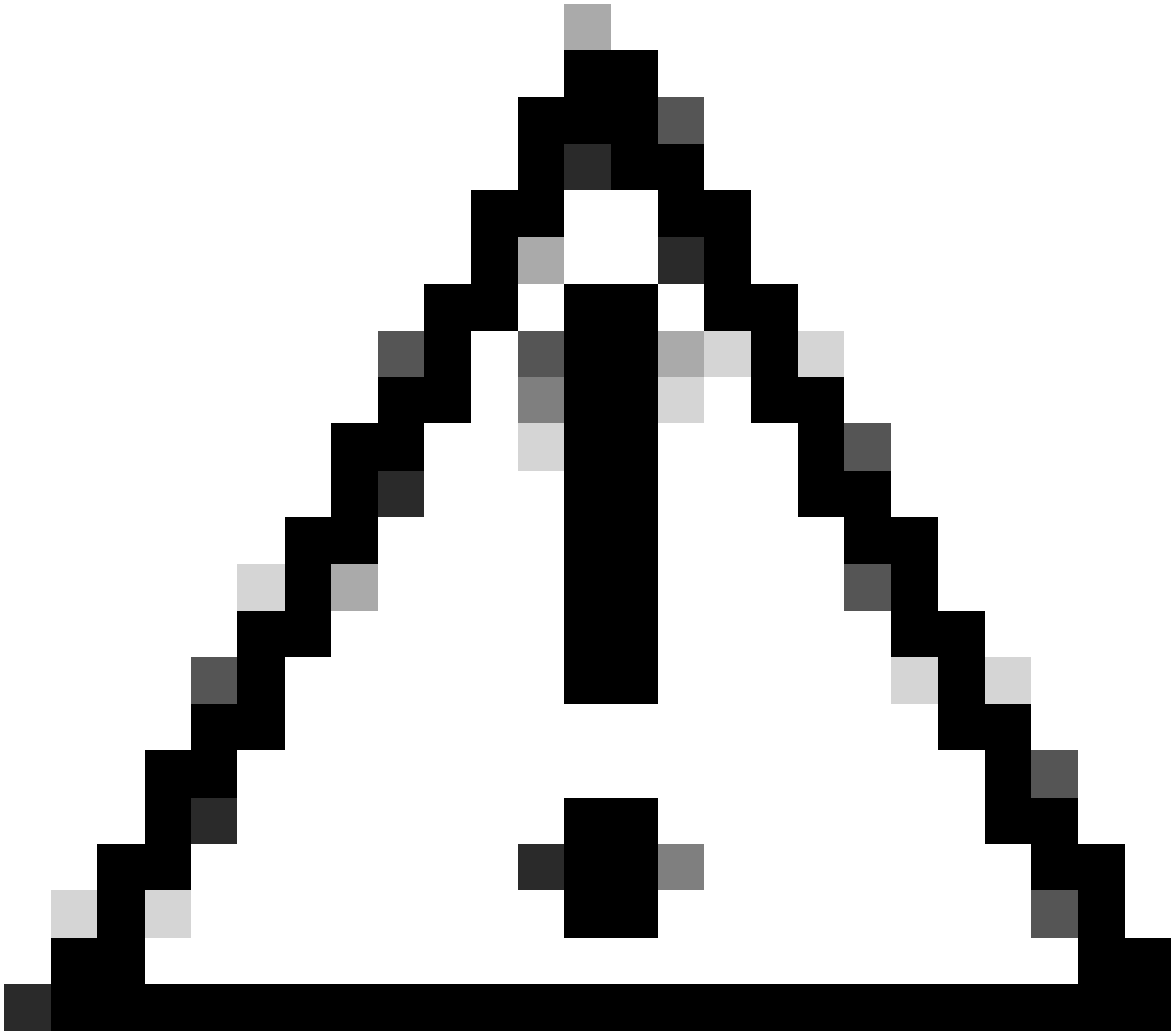
1단계. 클라이언트 신뢰 지점을 생성합니다.

```
cat9k(config)#crypto pki trustpoint svs_cat9k_25jun17
cat9k(ca-trustpoint)#revocation-check none
```

2단계. PKCS#12 파일을 bootflash에 복사합니다.



참고: PKCS#12 파일에 전체 인증서 체인과 개인 키가 암호화된 파일로 포함되어 있는지 확인합니다.



주의: 가져온 PKCS#12의 키는 RSA여야 합니다(예: ECC가 아닌 RSA 2048) 유형

<#root>

cat9k#

copy sftp bootflash: vrf Mgmt-vrf

Address or name of remote host [10.225.253.247]?
Source username [svs-user]?
Source filename [cat9k.svs.lab.pfx]? /home/svs-user/upload/cat9k-25jun17.pfx
Destination filename [cat9k-25jun17.pfx]?
Password:
!
2960 bytes copied in 3.022 secs (979 bytes/sec)

3단계. import 명령을 사용하여 PKCS#12 파일을 가져옵니다.

<#root>

```
cat9k#

crypto pki import svs_cat9k_25jun17 pkcs12 bootflash:cat9k-25jun17.pfx

 password C1sco.123
% Importing pkcs12...Reading file from bootflash:cat9k-25jun17.pfx
CRYPTO_PKI: Imported PKCS12 file successfully.
cat9k#

cat9k#

show crypto pki certificates svs_cat9k_25jun17


Certificate
  Status: Available
  Certificate Serial Number (hex): 5860BF33A2033365
  Certificate Usage: General Purpose
  Issuer:
    cn=SVS LabCA
    ou=SVS
    o=Cisco
    l=Raleigh
    st=North Carolina
    c=US
  Subject:
    Name: cat9k.svs.lab
    e=pkalkur@cisco.com
    cn=cat9k.svs.lab
    ou=svs
    o=cisco
    l=rtp
    st=nc
    c=us
  Validity Date:
    start date: 17:56:00 UTC Jun 17 2025
    end   date: 17:56:00 UTC Jun 17 2026
  Associated Trustpoints: svs_cat9k_25jun17

CA Certificate
  Status: Available
  Certificate Serial Number (hex): 20CD7402C4DA37F5
  Certificate Usage: General Purpose
  Issuer:
    cn=SVS LabCA
    ou=SVS
    o=Cisco
    l=Raleigh
    st=North Carolina
    c=US
  Subject:
    cn=SVS LabCA
    ou=SVS
    o=Cisco
    l=Raleigh
    st=North Carolina
    c=US
  Validity Date:
    start date: 17:05:00 UTC Apr 28 2025
    end   date: 17:05:00 UTC Apr 28 2035
  Associated Trustpoints: svs_cat9k_25jun17 svs_cat9k
  Storage: nvram:SVSLabCA#37F5CA.cer
```

TACACS 및 AAA(TLS 컨피그레이션 포함)

1단계. TACACS 서버 및 AAA 그룹을 생성하고 클라이언트(라우터) 신뢰 지점을 연결합니다.

```
tacacs server svs_tacacs
 address ipv4 10.225.253.209
 single-connection
 tls port 6049
 tls idle-timeout 60
 tls connection-timeout 60
 tls trustpoint client svs_cat9k
 tls ip tacacs source-interface GigabitEthernet0/0
 tls ip vrf forwarding Mgmt-vrf
!
aaa group server tacacs+ svs_tls
 server name svs_tacacs
 ip vrf forwarding Mgmt-vrf
!
tacacs-server directed-request
```

2단계. AAA 방법을 구성합니다.

```
aaa authentication login default group svs_tls local enable
aaa authentication login console local enable
aaa authentication enable default group svs_tls enable
aaa authorization config-commands
aaa authorization exec default group svs_tls local if-authenticated
aaa authorization commands 1 default group svs_tls local if-authenticated
aaa authorization commands 15 default group svs_tls
aaa accounting exec default start-stop group svs_tls
aaa accounting commands 1 default start-stop group svs_tls
aaa accounting commands 15 default start-stop group svs_tls
aaa session-id common
```

확인

구성 확인.

```
show tacacs
show crypto pki certificates <>
show crypto pki trustpoints <>
```

AAA 및 TACACS+에 대한 디버그

```
debug aaa authentication
debug aaa authorization
debug aaa accounting
debug aaa subsys
debug aaa protocol local
debug tacacs authentication
debug tacacs authorization
debug tacacs accounting
debug tacacs events
debug tacacs packet
debug tacacs
debug tacacs secure

! Below debugs will be needed only if there is any issue with SSL Handshake
debug ip tcp transactions
debug ip tcp packet
debug crypto pki transactions
debug crypto pki API
debug crypto pki messages
debug crypto pki server
debug ssl openssl errors
debug ssl openssl msg
debug ssl openssl states
clear logging
```

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.