

Nexus 9000 플랫폼에서 POAP 구현 및 확인

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[토폴로지](#)

[배경 정보](#)

[수동 프로비저닝 대 POAP](#)

[상위 레벨 POAP 워크플로](#)

[POAP 아키텍처 및 워크플로](#)

[파워-업 단계](#)

[USB 검색 단계](#)

[DHCP 검색 단계](#)

[DHCP 교환](#)

[DHCP를 통해 얻은 정보](#)

[DHCP 리스 및 오퍼 검증](#)

[POAP 디바이스 식별](#)

[POAP 스크립트 검색](#)

[POAP 스크립트 무결성](#)

[스크립트 실행 단계](#)

[소프트웨어 이미지 처리](#)

[설치 후 다시 로드 단계](#)

[구성 재생](#)

[POAP 요구 사항](#)

[POAP 구성](#)

[프로비저닝 인프라 요구 사항](#)

[POAP 스크립트 준비](#)

[디바이스 ID 구성](#)

[프로비저닝 로직 예](#)

[디바이스 컨피그레이션 준비](#)

[관리 액세스 고려 사항](#)

[파일 무결성 정보 준비](#)

[POAP 스크립트 MD5 업데이트](#)

[컨피그레이션 파일 MD5 준비](#)

[Cisco NX-OS 이미지 MD5 준비](#)

[POAP 시작](#)

[시작 구성 제거](#)

[스위치 다시 로드](#)

[POAP 초기화](#)

[부트스트랩 네트워크 정보 얻기](#)

[POAP 스크립트 다운로드 및 시작](#)

[다음을 확인합니다.](#)

[POAP 초기화 확인](#)

[POAP 로그 파일 확인](#)

[DHCP 검색 확인](#)

[DHCP Exchange 확인](#)

[패킷 캡처로 DHCP 확인](#)

[POAP 스크립트 검색 확인](#)

[패킷 캡처로 스크립트 검색 확인](#)

[POAP 스크립트 무결성 확인](#)

[디바이스별 컨피그레이션 검색 확인](#)

[구성 파일 검색 확인](#)

[패킷 캡처로 컨피그레이션 검색 확인](#)

[컨피그레이션 무결성 확인](#)

[구성 처리 확인](#)

[이미지 처리 확인](#)

[이미지 MD5 검색 확인](#)

[소프트웨어 설치 워크플로 확인](#)

[구성 재생 확인](#)

[ASCII 컨피그레이션 재생 확인](#)

[POAP 완료 확인](#)

[프로비저닝된 컨피그레이션 확인](#)

[결론](#)

소개

이 문서에서는 Cisco NX-OS를 실행하는 Cisco Nexus 9000 스위치의 POAP(PowerOn Auto Provisioning)에 대해 설명합니다.

사전 요구 사항

요구 사항

Cisco에서는 다음 항목에 대한 기본 지식을 갖춘 것을 권장합니다.

- Cisco NX-OS CLI(Command Line Interface)
- 기본 IPv4(Internet Protocol version 4) 주소 지정 및 레이어 3 연결
- DHCP(Dynamic Host Configuration Protocol) 기본 사항
- 기본 HTTP(Hypertext Transfer Protocol) 및 TFTP(Trivial File Transfer Protocol) 개념

사용되는 구성 요소

구성 요소	플랫폼/소프트웨어	버전/값	목적
Nexus 스위치	N9K-C9336C-FX2	NX-OS 10.6(2)	POAP 장치
프로비저닝 서버	Ubuntu 서버	24.04.5리터	외부 프로비저닝 인프라
DHCP/TFTP 서비스	dnsmasq	2.91	DHCP 및 TFTP 서비스 제공

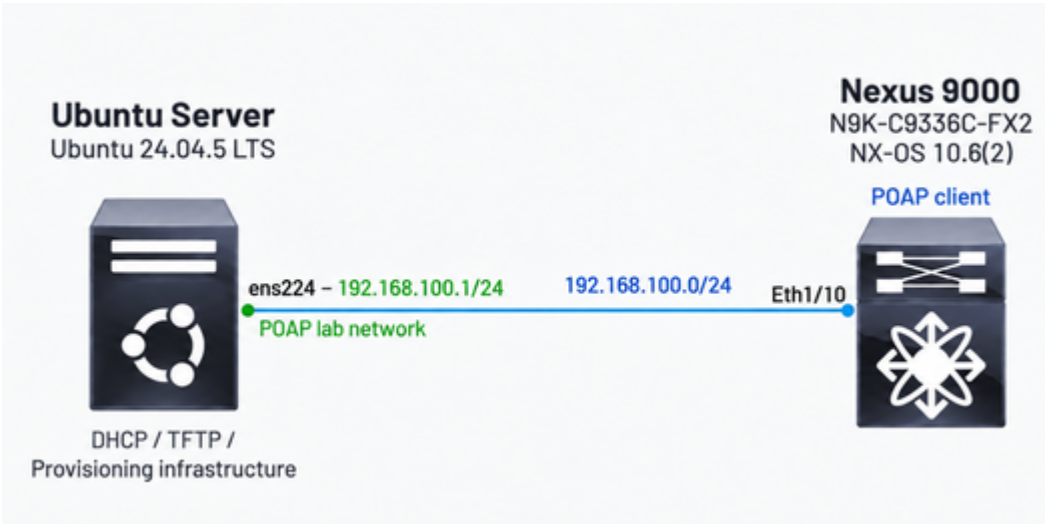
이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

토폴로지

이 실습에서는 외부 프로비저닝 인프라를 제공하는 Ubuntu 서버에 직접 연결된 Cisco Nexus 9000 스위치를 사용합니다.

Nexus 스위치와 프로비저닝 서버 간에 추가 네트워크 디바이스 없이 POAP 워크플로를 관찰할 수 있도록 토폴로지를 단순하게 유지하는 것이 목적입니다.

Nexus 스위치의 Ethernet1/10은 Ubuntu 서버의 ens224에 직접 연결됩니다.



배경 정보

새 Cisco Nexus 스위치를 구축하려면 디바이스가 네트워크에 참여하려면 초기 컨피그레이션이 필요합니다.

POAP(PowerOn Auto Provisioning)는 이 부트스트랩 프로세스를 자동화합니다. 지원되는 Nexus 스위치가 시작 컨피그레이션 없이 부팅되면 Cisco NX-OS는 POAP 모드로 들어가서 DHCP를 통해 임시 네트워크 정보를 얻고 프로비저닝 스크립트를 검색한 다음 해당 스크립트를 사용하여 필요한 소프트웨어 이미지 및 디바이스 컨피그레이션을 프로비저닝할 수 있습니다.

POAP는 주로 초기 또는 Day-0 프로비저닝과 관련이 있습니다. 스위치에서 CLI를 통해 전체 컨피그레이션을 수동으로 입력하지 않고도 구축에 필요한 리소스를 얻을 수 있기 때문입니다.

수동 프로비저닝 대 POAP

POAP가 없으면 Nexus 스위치를 처음 구축할 때 네트워크 연결을 구성하고, 구성 파일을 전송하고, 필요한 Cisco NX-OS 소프트웨어 이미지를 준비하기 위해 수동 상호 작용이 필요할 수 있습니다.

POAP를 통해 스위치는 프로비저닝 인프라를 자동으로 검색하고 구축에 필요한 리소스를 검색할 수 있습니다.

상위 레벨 POAP 워크플로

상위 레벨에서 POAP는 지원되는 Nexus 스위치가 시작 컨피그레이션 없이 부팅될 때 시작됩니다. 스위치는 프로비저닝 프로세스를 시작하고 프로비저닝 인프라에 액세스하는 데 필요한 네트워크 정보를 얻으려고 시도합니다.

네트워크 연결이 설정되면 스위치가 POAP 스크립트를 검색하여 실행합니다. 스크립트는 디바이스와 연결된 소프트웨어 이미지 및 컨피그레이션 리소스를 결정하고 나머지 프로비저닝 작업을 제어합니다.

Cisco는 5가지 기본 단계로 POAP 프로세스를 정의합니다.

- 전원 켜기
- USB 검색
- DHCP 검색
- 스크립트 실행
- 설치 후 다시 로드

다음 섹션에서는 프로비저닝 프로세스 중 각 단계의 역할에 대해 설명합니다.

POAP 아키텍처 및 워크플로

POAP는 Cisco NX-OS에서 단계별 부트스트래핑 프로세스로 작동합니다. 각 단계는 자동 디바이스 프로비저닝에 필요한 특정 기능을 수행합니다.

이러한 단계를 이해하면 이 문서의 뒷부분에 설명되어 있는 구현 및 확인 워크플로의 기초가 됩니다.

파워-업 단계

Nexus 스위치의 전원이 켜지면 Cisco NX-OS는 설치된 소프트웨어 이미지를 로드하고 시작 컨피그레이션을 검색합니다.

시작 컨피그레이션이 존재하는 경우 스위치는 정상 부팅 프로세스를 계속 진행합니다. 시작 컨피그레이션이 없으면 POAP 모드가 시작됩니다.

시작 중에 NX-OS는 POAP를 중단하고 일반적인 대화형 설정을 계속하기 위한 옵션을 제공하는 프롬프트를 표시합니다. POAP는 사용자의 개입 없이 계속됩니다.

POAP가 계속될 경우, 검색 프로세스가 시작되기 전에 전면 패널 인터페이스가 기본 컨피그레이션에 배치됩니다.

조건	결과
시작 컨피그레이션이 있습니다.	NX-OS는 정상적인 부팅 프로세스를 계속 진행합니다.
시작 컨피그레이션이 없습니다.	POAP 모드가 시작됩니다.
POAP가 수동으로 중단되었습니다.	NX-OS는 일반적인 대화형 설정으로 들어갑니다.
POAP가 계속됩니다.	이 프로세스는 USB 검색을 진행합니다.

USB 검색 단계

POAP가 시작되면 NX-OS는 액세스 가능한 USB 디바이스에서 프로비저닝 리소스를 검색합니다.

USB 디바이스에서 POAP 스크립트가 발견되면 NX-OS는 스크립트 실행을 시작합니다. 사용 가능한 스크립트가 없으면 POAP는 DHCP 검색을 진행합니다.

USB 기반 프로비저닝은 POAP 워크플로의 일부입니다. 그러나 이 문서에서는 DHCP를 통한 네트워크 기반 프로비저닝에 중점을 둡니다.

DHCP 검색 단계

USB 검색 중에 POAP 스크립트를 찾을 수 없으면 스위치에서 DHCP 검색을 시작합니다.

이 단계에서 스위치는 POAP에 필요한 정보를 제공할 수 있는 DHCP 서버를 찾기 위해 사용 가능한 전면 패널 인터페이스 또는 관리 인터페이스를 통해 DHCP Discover 메시지를 보냅니다.

Cisco Nexus DHCP 클라이언트는 클라이언트 식별자의 스위치 일련 번호를 사용합니다. 디바이스를 식별하고 프로비저닝 프로세스에 필요한 부트스트랩 정보를 제공하기 위해 프로비저닝 인프라에서 이 정보를 사용할 수 있습니다.

DHCP 교환

DHCP 협상은 표준 DORA(Discover, Offer, Request, Acknowledgment) 프로세스를 사용합니다.

DHCP 메시지	목적
검색	Nexus 스위치는 사용 가능한 DHCP 서버를 검색합니다.
제안	DHCP 서버는 네트워크 및 POAP 관련 정보를 제공합니다.
요청	스위치에서 선택한 DHCP 제공을 요청합니다.
확인 응답	DHCP 서버는 임대를 확인하고 IP 주소를 할당합니다.

DHCP를 통해 얻은 정보

네트워크 기반 POAP의 경우 DHCP 교환은 POAP 스크립트를 찾는 데 필요한 임시 네트워크 정보 및 부트스트랩 매개변수를 제공합니다.

DHCP Discover 메시지는 다음을 포함하는 정보를 요청합니다.

- TFTP 서버 이름 또는 주소
- 부팅 파일 이름

부트 파일 이름은 DHCP 협상이 완료된 후 스위치가 검색하는 POAP 스크립트를 식별합니다.

DHCP는 최종 Cisco Nexus 컨피그레이션을 제공하지 않습니다. 임시 연결을 설정하고 프로비저닝 스크립트를 찾는 데 필요한 부트스트랩 정보를 제공합니다.

DHCP 리스 및 오퍼 검증

POAP에는 최소 DHCP 임대 기간이 3600초가 필요합니다.

DHCP 제안을 받았다고 해서 DHCP 검색이 자동으로 성공하는 것은 아닙니다. 이 서비스는 스위치가 DHCP 요청 및 승인 교환을 계속하기 전에 POAP에서 예상되는 요구 사항을 충족해야 합니다.

유효한 오퍼가 수신되지 않으면 해당 오퍼를 통해 IP 주소가 할당되지 않습니다.

POAP 디바이스 식별

Cisco NX-OS Release 7.0(3)I7(4)부터 POAP는 RFC 3004에 정의된 DHCP 사용자 클래스 옵션을 지원합니다.

DHCPv4의 경우 POAP는 Cisco-POAP 값과 함께 사용자 클래스 옵션 77을 사용합니다.

이 값은 DHCP 요청을 POAP 프로세스에서 시작하는 것으로 식별합니다.

Cisco-POAP 값은 확인 과정에서 사용된 패킷 캡처에서 나중에 확인할 수 있습니다.

POAP 스크립트 검색

DHCP 협상이 완료되면 스위치는 DHCP 서버에서 받은 프로비저닝 정보를 사용하여 POAP 스크립트를 찾아 검색합니다.

이 문서에 사용된 표준 네트워크 기반 워크플로에서 DHCP 정보는 프로비저닝 스크립트가 포함된 TFTP 서버 및 부팅 파일을 식별합니다. 임시 네트워크 연결이 설정되면 Cisco NX-OS는 프로비저닝 서버에서 스크립트를 다운로드합니다.

POAP 스크립트에는 디바이스와 관련된 소프트웨어 이미지 및 컨피그레이션 리소스를 확인하고 처리하는 데 필요한 논리가 포함되어 있습니다.

Cisco에서는 Python으로 작성된 참조 POAP 스크립트를 제공합니다. 프로비저닝 환경과 일치하도록 스크립트를 사용자 지정할 수 있습니다.

POAP 스크립트 무결성

Cisco 참조 POAP 스크립트에는 MD5 체크섬이 포함되어 있습니다.

스크립트가 수정되면 스크립트가 POAP에 사용되기 전에 포함된 체크섬을 다시 계산해야 합니다.

체크섬 준비 절차는 파일 무결성 정보 준비 섹션의 뒷부분에서 설명합니다.

스크립트 실행 단계

POAP 스크립트가 다운로드되고 검증되면 Cisco NX-OS는 스위치에서 로컬로 스크립트를 실행합니다.

참조 스크립트는 다음과 같은 프로비저닝 작업을 수행할 수 있습니다.

- 스위치를 확인합니다.
- 디바이스별 컨피그레이션을 선택합니다.
- 컨피그레이션을 검색하고 확인합니다.
- 대상 Cisco NX-OS 이미지를 확인합니다.
- 소프트웨어 이미지를 검색하거나 검증합니다.
- 재생을 위한 컨피그레이션을 준비합니다.
- 프로비저닝 워크플로를 완료합니다.

다운로드한 컨피그레이션은 스크립트 실행 중에 반드시 바로 적용되는 것은 아닙니다.

Cisco NX-OS는 컨피그레이션을 스테이징하고 대상 소프트웨어 이미지를 사용하여 스위치가 부팅된 후 적용할 수 있습니다. 따라서 구성은 준비된 소프트웨어 환경에서 처리됩니다.

소프트웨어 이미지 처리

POAP 스크립트는 대상 Cisco NX-OS 이미지를 로컬에서 이미 사용할 수 있는지 여부를 확인할 수 있습니다.

필요한 이미지가 없는 경우 스크립트는 프로비저닝 인프라에서 이미지를 검색할 수 있습니다.

필요한 이미지를 이미 사용할 수 있는 경우 스크립트는 기존 파일을 검증하고 다른 복사본을 다운로드하지 않을 수 있습니다.

검증된 구현에서는 대상 이미지가 bootflash:에 이미 있습니다. 스크립트가 이미지의 유효성을 검사한 후 나머지 프로비저닝 작업을 계속합니다.

설치 후 다시 로드 단계

스크립트 실행이 완료되면 스위치가 설치 후 다시 로드 단계로 들어갑니다.

이 단계에서 소프트웨어 이미지 변경이 프로비저닝 프로세스에 포함된 경우 스위치가 다시 시작되고 대상 Cisco NX-OS 소프트웨어 이미지로 부팅됩니다.

그런 다음 스위치에서 다운로드한 컨피그레이션을 적용하거나 다시 재생합니다. 컨피그레이션이 성공적으로 적용되면 Cisco NX-OS는 실행 중인 컨피그레이션을 시작 컨피그레이션에 복사합니다.

이 순서를 통해 프로비저닝된 컨피그레이션이 준비된 소프트웨어 환경에서 처리됩니다.

구성 재생

컨피그레이션 재생은 스위치가 다시 시작된 후 Cisco NX-OS가 POAP 중에 다운로드한 컨피그레이션을 적용하는 프로세스입니다.

컨피그레이션 파일은 처음 다운로드할 때 반드시 적용할 필요는 없습니다. 대신 POAP는 스크립트 실행 중에 컨피그레이션을 스테이징하고 대상 Cisco NX-OS 이미지가 활성화된 후에 적용할 수 있습니다.

프로세스는 다음과 같이 요약할 수 있습니다.

단계	작업
스크립트 실행	필요한 소프트웨어 이미지 및 컨피그레이션을 검색합니다.
설치 후 다시 로드	프로비저닝 작업 후 스위치를 다시 시작합니다.
구성 재생	다운로드한 구성을 적용합니다.
구성 지속성	실행 중인 컨피그레이션을 시작 컨피그레이션에 복사합니다.

컨피그레이션 검색과 컨피그레이션 애플리케이션 간의 이러한 분리는 POAP 워크플로에서 중요한 부분입니다.

POAP 요구 사항

네트워크 기반 POAP는 임시 네트워크 연결 및 프로비저닝 리소스에 대한 액세스를 제공하기 위해 외부 인프라가 필요합니다.

표준 POAP 워크플로의 경우 필요한 인프라는 다음과 같습니다.

- 부트스트랩 중에 필요한 네트워크 정보를 제공하는 DHCP 서버입니다.
- POAP 스크립트를 제공하는 TFTP 서버입니다.
- 필요한 Cisco NX-OS 소프트웨어 이미지 및 디바이스 컨피그레이션 파일을 제공하는 하나 이상의 서버.

DHCP를 통해 얻은 네트워크 정보는 POAP 프로비저닝 프로세스 중에 사용됩니다.

POAP 구성

이 섹션에서는 Cisco Nexus 9000 스위치에 네트워크 기반 POAP(PowerOn Auto Provisioning)를 구현하는 데 필요한 구성 요소에 대해 설명합니다.

POAP는 임시 네트워크 연결 및 프로비저닝 리소스에 대한 액세스를 제공하기 위해 외부 DHCP 및 파일 전송 서비스를 사용합니다. 서드파티 서버 플랫폼의 구성은 이 문서의 범위에 속하지 않습니다.

이 섹션의 예에서는 POAP 구현을 시연하기 위해 앞서 설명한 토폴로지를 사용합니다.

프로비저닝 인프라 요구 사항

검증된 구현을 위해 프로비저닝 인프라는 POAP 동안 Nexus 스위치에 필요한 임시 네트워크 정보 및 파일 전송 서비스를 제공합니다.

DHCP 인프라는 임시 연결을 설정하고 POAP 스크립트를 찾기 위해 필요한 정보를 제공해야 합니다. POAP에는 최소 3600초의 DHCP 리스가 필요합니다.

정보	목적
IP 주소	POAP 중에 임시 네트워크 연결을 제공합니다.
서브넷 마스크	임시 IP 서브넷을 정의합니다.
기본 게이트웨이	라우팅이 필요할 때 리소스를 프로비저닝하는 연결을 제공합니다.
TFTP 서버 이름 또는 주소	POAP 스크립트가 검색되는 서버를 식별합니다.
부팅 파일 이름	Nexus 스위치가 다운로드하는 POAP 스크립트를 식별합니다.

유효한 DHCP 오퍼를 선택한 후 Nexus 스위치는 DHCP 협상을 완료하고 프로비저닝 프로세스 중에 할당된 네트워크 정보를 사용합니다.

이 문서의 검증된 예에서 프로비저닝 인프라는 다음 값을 사용합니다.

매개변수	실습 가치
POAP 네트워크	192.168.100.0/24
프로비저닝 서버	192.168.100.1
DHCP 주소 범위	192.168.100.100–192.168.100.110
DHCP 임대	2시간
TFTP 서버	192.168.100.1
POAP 스크립트	poap.py
Nexus 인터페이스	이더넷1/10

이 예에서 사용되는 프로비저닝 서버는 Ubuntu를 실행하며 구현에 필요한 DHCP 및 TFTP 서비스를 제공합니다. 서드파티 서버 소프트웨어의 세부 구성은 이 문서의 범위에 속하지 않습니다.

POAP 스크립트 준비

Cisco는 Python에서 개발한 참조 POAP 스크립트를 제공합니다. 프로비저닝 환경과 일치하도록 사용자가 편집할 수 있는 옵션을 수정합니다.

POAP 스크립트는 프로비저닝 오케스트레이터의 역할을 합니다. Nexus 스위치가 스크립트를 다운로드하고 검증한 후 스크립트는 스위치가 프로비저닝에 필요한 컨피그레이션 및 소프트웨어 이미지 리소스를 검색하고 처리하는 방법을 결정합니다.

검증된 구현의 경우 다음 옵션이 사용됩니다.

```
options = { "username": "root", "password": "password", "hostname": "192.168.100.1", "transfer_protocol": "tftp", "mode": "serial_number",
"target_system_image": "nxos64-cs.10.6.2.F.bin", "config_path": "/", "target_image_path": "/", }
```

이 예에 표시된 값은 실습 환경에 해당합니다. 프로덕션 가치는 프로비저닝 인프라 및 구축 요구 사항에 따라 달라집니다.

옵션	실습 가치	목적
호스트 이름	192.168.100.1	프로비저닝 파일이 검색되는 서버를 식별합니다.
전송 프로토콜	tftp	파일 전송 프로토콜을 정의합니다.
모드로 들어갑니다	일련 번호	디바이스별 컨피그레이션 파일을 선택하기 위해 스위치 일련 번호를 사용합니다.
대상_시스템_이미지	nxos64-cs.10.6.2.F.bin	대상 Cisco NX-OS 이미지를 정의합니다.
구성_경로	/	디바이스 컨피그레이션을 포함하는 경로를 정의합니다.
대상 이미지 경로	/	대상 소프트웨어 이미지를 포함하는 경로를 정의합니다.

참조 스크립트에는 원격 서버에서 파일을 검색할 때 호스트 이름, 사용자 이름 및 비밀번호가 필요합니다. 이 예에서는 사용자 이름 및 비밀번호 인증을 제공하지 않는 TFTP를 사용합니다. 그러나 참조 스크립트는 초기화 중에 이러한 매개변수가 있는지 계속 확인합니다.

디바이스 ID 구성

검증된 구현에서는 다음을 사용합니다.

```
"mode": "serial_number"
```

serial_number 모드에서 스크립트는 로컬 스위치의 일련 번호를 검색하여 해당 컨피그레이션 파일 이름을 확인하는 데 사용합니다.

예를 들면 다음과 같습니다.

스위치 일련 번호: FDO 262921L3

구성 파일: conf.FDO262921L3

구성 파일 이름에는 conf.<serial-number> 형식을 사용합니다.

따라서 동일한 poap.py 스크립트는 여러 Nexus 스위치를 지원할 수 있으며 각 스위치는 디바이스별 컨피그레이션을 검색합니다.

프로비저닝 로직 예

상위 레벨에서 검증된 스크립트는 다음 시퀀스를 수행합니다.

Identify the switch. ↓ Select conf.

↓ Retrieve and validate the configuration. ↓ Determine the target Cisco NX-OS image. ↓ Validate or retrieve the image. ↓ Prepare the configuration for replay. ↓ Complete the provisioning workflow.

POAP 스크립트에 전체 디바이스 컨피그레이션이 포함되어 있지 않습니다. 여기에는 프로비저닝 리소스를 찾고 검증하고 처리하는 데 필요한 논리와 매개변수가 포함되어 있습니다.

디바이스 컨피그레이션 준비

디바이스 컨피그레이션 파일에는 POAP가 대상 Nexus 스위치에 프로비저닝하는 Cisco NX-OS 컨피그레이션이 포함되어 있습니다.

이 예에서 사용되는 스위치의 serial_number 모드는 이 컨피그레이션 파일을 선택합니다.

일련 번호: FDO 262921L3
구성 파일: conf.FDO262921L3

컨피그레이션 파일에는 의도한 Day-0 컨피그레이션을 설정하는 데 필요한 명령이 포함되어 있습니다.

검증된 구현의 경우 컨피그레이션에는 관리 사용자, 호스트 이름 및 루프백 인터페이스가 포함됩니다.

```
username admin password
```

```
role network-admin hostname N9K-POAP-SW1 interface loopback100 description POAP-VALIDATION ip address 10.255.255.100/32 no shutdown
```

프로비저닝 워크플로의 검증을 간소화하기 위해 컨피그레이션을 의도적으로 최소화합니다.

POAP가 완료되면 호스트 이름, 관리 계정 및 루프백100 인터페이스는 디바이스별 컨피그레이션이 성공적으로 적용되었음을 나타내는 명확한 표시기를 제공합니다.

관리 액세스 고려 사항

프로비저닝된 컨피그레이션에는 POAP가 완료된 후 필요한 관리 액세스가 포함되어야 합니다.



주의: 디바이스 컨피그레이션에 유효한 관리자 사용자 및 비밀번호가 포함되어 있는지 확인합니다. 프로비저닝된 컨피그레이션이 유효한 관리 액세스를 제공하지 않더라도 POAP가 성공적으로 완료될 수 있습니다.

초기 Lab 검증 과정에서 디바이스 컨피그레이션에 호스트 이름과 Loopback100 인터페이스가 포함되었지만 관리 사용자는 포함되지 않았습니다. POAP가 성공적으로 완료되었습니다. 그러나 프로비저닝 후에는 관리 액세스를 사용할 수 없었습니다.

POAP를 완료하면 프로비저닝 워크플로가 성공적으로 완료되었음을 확인합니다. Day-0 컨피그레이션에 스위치를 작동 및 액세스하는 데 필요한 모든 설정이 포함되어 있는지 확인하지는 않습니다.

파일 무결성 정보 준비

참조 POAP 워크플로에서는 Nexus 스위치에서 처리하기 전에 프로비저닝 리소스의 무결성을 확인하기 위해 MD5 체크섬을 사용합니다.

POAP는 예상 체크섬을 해당 파일에서 계산된 체크섬과 비교합니다. 값이 일치하면 파일에 예상 데이터가 포함되어 있는지 확인합니다.

검증된 구현에서는 다음 리소스에 대해 무결성 정보를 사용합니다.

리소스	무결성 정보	목적
POAP 스크립트	poap.py에 내장된 MD5 값	실행 전에 스크립트를 검증합니다.
디바이스 컨피그레이션	conf.FDO262921L3.md5	다운로드한 디바이스 컨피그레이션을 확인합니다.
Cisco NX-OS 이미지	nxos64-cs.10.6.2.F.bin.md5	대상 소프트웨어 이미지를 검증합니다.

참조 스크립트의 disable_md5 옵션은 MD5 검증을 제어합니다. 기본값은 False입니다.

POAP 스크립트 MD5 업데이트

참조 POAP 스크립트에는 파일의 시작 부분에 MD5 값이 포함되어 있습니다.

```
#md5sum="
"
```

스크립트 내용을 수정한 후 POAP에 스크립트를 사용하려면 먼저 포함된 체크섬을 다시 계산해야 합니다.

이 구현에 사용된 스크립트의 경우 체크섬이 다음으로 업데이트되었습니다.

```
f=poap.py cat "$f" | sed '/^#md5sum/d' > "$f.md5" sed -i 's/^#md5sum=.*#md5sum=\"$f.md5\" | sed 's/./"/' "$f" rm -f "$f.md5"
```

결과 스크립트에는 다음이 포함됩니다.

```
#md5sum="e4dd91fd4296d40abd2f6187152cd86e"
```

POAP 중에 Cisco NX-OS는 다운로드한 스크립트의 체크섬을 계산하고 스크립트 실행 전에 포함된 값과 비교합니다.

컨피그레이션 파일 MD5 준비

참조 스크립트는 디바이스 컨피그레이션을 확인하기 위해 별도의 MD5 파일을 사용할 수 있습니다.

이 구현의 경우

구성 파일:

conf.FDO262921L3

무결성 파일:

conf.FDO262921L3.md5

무결성 파일에는 conf.FDO262921L3의 현재 버전과 연결된 체크섬이 포함되어야 합니다.

컨피그레이션이 다운로드되면 스크립트는 체크섬을 계산하고 컨피그레이션을 처리하기 전에 예상한 값과 비교합니다.

디바이스 컨피그레이션이 변경되면 다음 POAP 시도 전에 해당 MD5 정보를 업데이트합니다.

Cisco NX-OS 이미지 MD5 준비

참조 스크립트는 대상 Cisco NX-OS 이미지의 유효성을 검사할 수도 있습니다.

이 구현의 경우

대상 이미지:

nxos64-cs.10.6.2.F.bin

무결성 파일:

nxos64-cs.10.6.2.F.bin.md5

무결성 파일에는 다음이 포함됩니다.

ce50c3f0e387bee578e615a373a4ede0 nxos64-cs.10.6.2.F.bin

검증된 구현에서는 대상 이미지가 bootflash:에 이미 있습니다. POAP 스크립트는 기존 이미지의 MD5 체크섬을 계산하여 프로비저닝 서버에서 얻은 값과 비교했습니다.

두 값이 일치하므로 이미지 파일을 다시 다운로드할 필요가 없습니다.



참고: MD5 검증에서 파일 무결성을 확인합니다. 프로비저닝 파일을 암호화하거나 자체적으로 인증을 제공하지 않습니다.

POAP 시작

스위치에 시작 컨피그레이션이 없고 시스템 전체 POAP가 활성화된 경우 시스템 시작 중에 POAP가 실행됩니다.

프로비저닝 프로세스가 시작되기 전에 POAP 상태를 확인합니다.

```
N9K-POAP-SW1# show system poap System-wide POAP is enabled
```

Cisco NX-OS는 POAP가 시스템 없이 전역적으로 비활성화되도록 허용합니다. POAP를 다시 활성화하려면 시스템 poap를 사용하거나 POAP 비활성화 플래그와 쓰기 지우기 poap를 지웁니다.

시작 구성 제거

시작 컨피그레이션 없이 스위치가 부팅될 때 표준 POAP가 시작됩니다.

제어된 랩 환경에서 다음을 실행합니다.

```
switch# write erase
```



경고: 이 명령은 startup-configuration을 지웁니다.

Do you wish to proceed anyway? (y/n) [n] y

기본 쓰기 지우기 작업은 영구 메모리에서 시작 컨피그레이션을 제거합니다. 실행 중인 컨피그레이션 또는 부팅 변수가 지워지지 않습니다.

시작 컨피그레이션이 더 이상 없는지 확인합니다.

switch# show startup-config No startup configuration



주의: write erase(쓰기 지우기)는 저장된 스위치 컨피그레이션을 제거합니다. 컨피그레이션을 안전하게 제거 또는 복원할 수 있는 경우에만 이 절차를 사용합니다.

POAP 후 관리 액세스 권한을 잃지 않으려면 이전 섹션에서 준비한 디바이스 컨피그레이션에 필요한 관리 자격 증명이 포함되어 있는지 확인하십시오.

스위치 다시 로드

시작 컨피그레이션이 제거된 후 스위치를 다시 로드합니다.

Cisco NX-OS에서 현재 실행 중인 컨피그레이션을 저장하라는 프롬프트가 표시되면 저장하지 마십시오. 실행 중인 컨피그레이션을 저장하면 시작 컨피그레이션이 다시 작성되므로 표준 POAP 트리거 조건이 적용되지 않습니다.

시작 중에 Cisco NX-OS는 POAP를 초기화하고 프로세스를 중단하는 옵션을 제공합니다.

Abort Power On Auto Provisioning [yes - continue with normal setup, no - continue with Power On Auto Provisioning] (yes/no)[no]:

POAP를 계속하려면 기본 no 옵션을 수락합니다.

POAP 초기화

POAP가 초기화된 후 스위치는 먼저 USB 기반 프로비저닝 리소스를 확인합니다. 사용 가능한 USB 프로비저닝 스크립트가 없으면 POAP는 네트워크 기반 검색을 진행합니다.

검증된 구현으로 다음 메시지가 생성되었습니다.

```
2026 Sep 22 18:21:23 switch %$ VDC-1 %$ %POAP-2-POAP_INITED: [FDO262921L3-88:FC:5D:BA:D7:7F] - POAP process initialized
2026 Sep 22 18:21:41 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - USB Initializing Success 2026
Sep 22 18:21:41 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - USB disk not detected
```

그런 다음 POAP가 IPv4 DHCP 검색을 시작합니다.

```
2026 Sep 22 18:23:54 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Start DHCP v4 session 2026
Sep 22 18:23:54 switch %$ VDC-1 %$ %POAP-2-POAP_DHCP_DISCOVER_START: [FDO262921L3-88:FC:5D:BA:D7:7F] - POAP DHCP
Discover phase started
```

부트스트랩 네트워크 정보 얻기

유효한 DHCP 응답이 수신되면 Cisco NX-OS는 임시 네트워크 매개변수 및 POAP 스크립트의 위치를 표시합니다.

검증된 구현은 다음과 같은 결과를 낳았습니다.

```
2026 Sep 22 18:23:58 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: Recieved DHCP offer from server ip - 192.168.100.1 2026 Sep 22
18:24:02 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Using DHCP, valid information received
over Eth1/10 from 192.168.100.1 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F]
- Assigned IP address: 192.168.100.108 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-
88:FC:5D:BA:D7:7F] - Netmask: 255.255.255.0 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-
88:FC:5D:BA:D7:7F] - Default Gateway: 192.168.100.1 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-POAP_INFO:
[FDO262921L3-88:FC:5D:BA:D7:7F] - Script Server: 192.168.100.1 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-POAP_INFO:
[FDO262921L3-88:FC:5D:BA:D7:7F] - Script Name: poap.py
```

이 단계에서 DHCP 검색이 완료되었습니다. 스위치에는 임시 네트워크 연결 및 POAP 스크립트를 찾는 데 필요한 정보가 있습니다.

POAP 스크립트 다운로드 및 시작

스위치는 DHCP 검색 중에 식별된 POAP 스크립트를 검색합니다.

```
2026 Sep 22 18:24:14 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - The POAP Script download has
started 2026 Sep 22 18:24:14 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - The POAP Script is
being downloaded from [copy tftp://192.168.100.1/poap.py bootflash:scripts/script.sh vrf default ] 2026 Sep 22 18:24:15 switch %$ VDC-1 %$
%POAP-2-POAP_SCRIPT_DOWNLOADED: [FDO262921L3-88:FC:5D:BA:D7:7F] - Successfully downloaded POAP script file
```

스크립트 검색이 완료되면 POAP는 스크립트 검증 및 실행을 진행합니다. 그런 다음 참조 스크립트는 디바이스 식별, 컨피그레이션 검색, 소프트웨어 이미지 처리, 컨피그레이션 스테이징 및 나머지 프로비저닝 작업을 제어합니다.

다음을 확인합니다.

POAP 워크플로의 각 주요 단계를 확인하려면 이 섹션의 정보를 사용합니다.

POAP는 여러 관점에서 검증할 수 있다. Nexus 콘솔 및 POAP 로그는 내부 프로비저닝 프로세스에 대한 정보를 제공하며, DHCP 서버 로그 및 패킷 캡처는 네트워크 교환에 대한 추가 가시성을 제공합니다.

검증된 구현에서는 이러한 소스를 사용하여 프로비저닝 워크플로의 상관관계를 분석합니다.

확인 소스	제공된 정보
Nexus 콘솔	POAP 초기화, DHCP 검색, 스크립트 실행, 다시 로드 및 컨피그레이션 재생
Nexus POAP 로그	자세한 초기화 및 스크립트 실행 정보
DHCP 서버 로그	DHCP 협상 및 파일 전송 활동
패킷 캡처	DHCP 및 TFTP 교환
구성 실행 및 시작	최종 프로비저닝 결과



참고: 이 섹션의 출력은 제어된 POAP 검증 시도 중에 수집되었습니다. 관리 액세스 권한이 있는 최종 Day-0 컨피그레이션을 검증하기 위해 이후 시도가 사용되었습니다. 원본 타임스탬프와 로그 메시지는 예제에서 유지됩니다.

서드파티 서버 로그 및 패킷 캡처는 추가 확인 소스로만 포함됩니다. 타사 소프트웨어의 구성은 이 문서의 범위에 속하지 않습니다.

POAP 초기화 확인

시스템 시작 중에 POAP가 성공적으로 초기화되었는지 확인합니다.

Nexus 콘솔에 초기화 이벤트가 표시됩니다.

```
2026 Sep 22 18:21:23 switch %$ VDC-1 %$ %POAP-2-POAP_INITED: [FDO262921L3-88:FC:5D:BA:D7:7F] - POAP process initialized
```

이 메시지는 스위치가 POAP 워크플로에 들어갔음을 확인합니다.

그런 다음 POAP는 USB 기반 프로비저닝 리소스를 확인합니다.

```
2026 Sep 22 18:21:41 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - USB Initializing Success 2026
Sep 22 18:21:41 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - USB disk not detected
```

이 예에서는 사용 가능한 USB 프로비저닝 리소스가 없습니다. 따라서 POAP는 네트워크 기반 검색을 계속합니다.

그러면 스위치가 DHCP 검색을 시작합니다.

```
2026 Sep 22 18:23:54 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Start DHCP v4 session 2026
Sep 22 18:23:54 switch %$ VDC-1 %$ %POAP-2-POAP_DHCP_DISCOVER_START: [FDO262921L3-88:FC:5D:BA:D7:7F] - POAP DHCP
Discover phase started
```

이러한 메시지는 USB 검색에서 네트워크 기반 DHCP 검색으로의 전환을 확인합니다.

POAP 로그 파일 확인

POAP는 bootflash:에 초기화 및 스크립트 실행 로그를 저장합니다. 이러한 파일은 콘솔 이벤트가 생성된 후 프로비저닝 프로세스에 대한 추가 정보를 제공합니다.

```
N9K-POAP-SW1# dir bootflash: | include poap 5664 Sep 22 19:52:10 2026 20260922195156_poap_24710_script.log 87366 Sep 22 19:58:48
2026 20260922_194902_poap_24710_init.log N9K-POAP-SW1#
```

_init.log 파일에는 POAP 초기화 및 검색과 관련된 정보가 포함되어 있습니다.

_script.log 파일에는 POAP 스크립트를 실행하는 동안 생성된 자세한 정보가 들어 있습니다.

여러 POAP 시도가 있을 경우 각 파일을 해당 프로비저닝 시도와 연관시키려면 파일 이름의 타임스탬프를 사용합니다.

DHCP 검색 확인

POAP가 DHCP 검색 단계로 들어가면 Nexus 스위치가 유효한 부트스트랩 네트워크 정보를 수신하는지 확인합니다.

Nexus 콘솔에서 DHCP 검색이 시작되었고 Ethernet1/10을 통해 유효한 정보를 받았음을 확인합니다.

```
2026 Sep 22 18:23:54 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Start DHCP v4 session 2026
Sep 22 18:23:54 switch %$ VDC-1 %$ %POAP-2-POAP_DHCP_DISCOVER_START: [FDO262921L3-88:FC:5D:BA:D7:7F] - POAP DHCP
Discover phase started 2026 Sep 22 18:23:58 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: Recieved DHCP offer from server ip -
192.168.100.1 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Using DHCP,
valid information received over Eth1/10 from 192.168.100.1 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-POAP_INFO:
[FDO262921L3-88:FC:5D:BA:D7:7F] - Assigned IP address: 192.168.100.108 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-
POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Netmask: 255.255.255.0 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-
POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Default Gateway: 192.168.100.1 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-
2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Script Server: 192.168.100.1 2026 Sep 22 18:24:02 switch %$ VDC-1 %$ %POAP-2-
POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Script Name: poap.py
```

이 출력은 POAP가 Ethernet1/10을 선택하고 임시 주소 192.168.100.108을 얻었으며 다음 프로비저닝 단계에 필요한 게이트웨이, 스크립트 서버 및 스크립트 파일 이름을 수신했음을 확인합니다.

DHCP Exchange 확인

DHCP 서버 로그는 부트스트랩 교환의 추가 보기를 제공합니다.

검증된 예는 다음과 같습니다.

```
vendor class: Cisco N9K-C9336C-FX2 user class: Cisco-POAP DHCPDISCOVER(ens224) 88:fc:5d:ba:d7:7f DHCPOFFER(ens224)
192.168.100.108 88:fc:5d:ba:d7:7f DHCPREQUEST(ens224) 192.168.100.108 88:fc:5d:ba:d7:7f DHCPACK(ens224) 192.168.100.108
```

88:fc:5d:ba:d7:7f

서버는 POAP 부트스트랩 정보도 식별했습니다.

```
option: 67 bootfile-name poap.py option:150 tftp-server-address 192.168.100.1 option: 3 router 192.168.100.1
```

로그는 완전한 Discover, Offer, Request 및 Acknowledgment 교환을 확인합니다. 또한 로그는 프로비저닝 요청과 관련된 Cisco-POAP 사용자 클래스, 스크립트 파일 이름, TFTP 서버 주소 및 기본 게이트웨이를 표시합니다.

패킷 캡처로 DHCP 확인

패킷 캡처를 사용하여 네트워크에서 직접 DHCP 교환을 확인할 수 있습니다.

검증된 예에서는 다음과 같이 캡처를 검토했습니다.

```
sudo tcpdump -nn -vvv -r /tmp/poap-run1.pcap \ 'udp port 67 or udp port 68'
```

캡처 결과:

```
DHCP-Message (53), length 1: Discover Vendor-Class (60), length 20: "Cisco N9K-C9336C-FX2" DHCP-Message (53), length 1: Offer  
DHCP-Message (53), length 1: Request Requested-IP (50), length 4: 192.168.100.108 DHCP-Message (53), length 1: ACK
```

패킷 캡처는 DHCP 서버 로그에서 관찰된 Discover, Offer, Request 및 Acknowledgment 시퀀스를 확인합니다.

POAP 스크립트 검색 확인

DHCP 검색이 완료되면 Nexus 스위치가 DHCP 교환 중에 식별된 서버에서 POAP 스크립트를 검색하는지 확인합니다.

Nexus 콘솔은 스크립트 다운로드가 언제 시작되는지를 나타냅니다.

```
2026 Sep 22 18:24:14 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - The POAP Script download has
started 2026 Sep 22 18:24:14 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - The POAP Script is
being downloaded from [copy tftp://192.168.100.1/poap.py bootflash:scripts/script.sh vrf default ] 2026 Sep 22 18:24:15 switch %$ VDC-1 %$
%POAP-2-POAP_SCRIPT_DOWNLOADED: [FDO262921L3-88:FC:5D:BA:D7:7F] - Successfully downloaded POAP script file
```

이 메시지는 스위치가 DHCP를 통해 검색된 TFTP 서버에서 poap.py를 검색했음을 확인합니다.

패킷 캡처로 스크립트 검색 확인

패킷 캡처는 스크립트 실행 중에 생성된 TFTP 요청을 직접 확인하는 기능을 제공합니다.

검증된 예에서는 다음을 실행합니다.

```
sudo tcpdump -nn -r /tmp/poap-run1.pcap 'udp dst port 69' reading from file /tmp/poap-run1.pcap, link-type EN10MB (Ethernet), snapshot
length 262144 13:38:17.012862 IP 192.168.100.108.58485 > 192.168.100.1.69: TFTP, length 16, RRQ "poap.py" octet 13:38:19.955108 IP
192.168.100.108.58550 > 192.168.100.1.69: TFTP, length 29, RRQ "conf.FDO262921L3.md5" octet 13:38:20.651491 IP
192.168.100.108.58680 > 192.168.100.1.69: TFTP, length 25, RRQ "conf.FDO262921L3" octet 13:38:21.737101 IP 192.168.100.108.58866 >
192.168.100.1.69: TFTP, length 35, RRQ "nxos64-cs.10.6.2.F.bin.md5" octet
```

첫 번째 요청에서는 임시 POAP 주소 192.168.100.108이 192.168.100.1의 프로비저닝 서버에 접속하여 poap.py를 요청했음을 확인합니다. 이후 요청에서는 디바이스 컨피그레이션 및 이미지 검증으로의 진행 상황을 보여줍니다.

POAP 스크립트 무결성 확인

다운로드한 스크립트가 실행되기 전에 Cisco NX-OS는 스크립트에 포함된 MD5 체크섬을 검증합니다.

검증된 예는 다음과 같습니다.

```
2026 Sep 22 18:24:15 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-88:FC:5D:BA:D7:7F] - Script file size 113107, MD5
checksum e4dd91fd4296d40abd2f6187152cd86e 2026 Sep 22 18:24:15 switch %$ VDC-1 %$ %POAP-2-POAP_INFO: [FDO262921L3-
88:FC:5D:BA:D7:7F] - MD5 checksum received from the script file is e4dd91fd4296d40abd2f6187152cd86e 2026 Sep 22 18:24:15 switch %$
VDC-1 %$ %POAP-2-POAP_SCRIPT_STARTED_MD5_VALIDATED: [FDO262921L3-88:FC:5D:BA:D7:7F] - POAP script execution
started(MD5 validated)
```

일치하는 값은 다운로드한 스크립트가 실행을 시작하기 전에 무결성 검사를 통과했음을 확인합니다.

이 단계에서 제어는 POAP 초기화에서 다운로드된 스크립트에 포함된 프로비저닝 로직으로 전달됩니다.

디바이스별 컨피그레이션 검색 확인

POAP 스크립트가 시작된 후 스크립트가 Nexus 스위치를 식별하고 해당 디바이스 컨피그레이션을 선택하는지 확인합니다.

serial_number 모드에서 참조 스크립트는 로컬 스위치 일련 번호를 사용하여 컨피그레이션 파일 이름을 결정합니다.

검증된 예는 다음과 같이 생성됩니다.

일련 번호를 기반으로 소스 cfg 파일 이름 설정

일련 번호 FDO262921L3

선택한 conf 파일 이름: conf.FDO262921L3

출력에서는 스크립트가 로컬 일련 번호를 식별하고 conf.FDO262921L3을 선택했음을 확인합니다.

구성 파일 검색 확인

컨피그레이션 파일 이름을 선택한 후 스크립트는 해당 MD5 정보를 검색합니다.

```
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: - Logfile name: /bootflash/20260922182417_poap_23243_script.log - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: - Found 1 POAP script logs - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: - Setting source cfg filename based-on serial number - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: - serial number FDO262921L3 - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: - Selected conf file name : conf.FDO262921L3 - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Verifying freespace in bootflash - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Free bootflash space is 106776296.0 KB - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Single image is set - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Copying config file - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Copying MD5 information - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Starting Copy of MD5. src=/conf.FDO262921L3.md5 dest = /bootflash/conf.FDO262921L3.md5 - script.sh
2026 Sep 22 18:24:17 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-
```

MAC[88:FC:5D:BA:D7:7F] - File transfer_protocol = tftp - script.sh

무결성 파일이 다운로드되어 bootflash에 저장됩니다. 2026 Sep 22 18:24:17 switch %\$ VDC-1 %\$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - *** Downloaded file is of size 51 *** - script.sh 2026 Sep 22 18:24:17 switch %\$ VDC-1 %\$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Renamed /bootflash/conf.FDO262921L3.md5.tmp to /bootflash/conf.FDO262921L3.md5 - script.sh

그런 다음 스크립트는 디바이스 컨피그레이션을 검색합니다.

2026 Sep 22 18:24:17 switch %\$ VDC-1 %\$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - INFO: Starting Copy of Config File to /bootflash/poap_conf.cfg - script.sh 2026 Sep 22 18:24:17 switch %\$ VDC-1 %\$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Copying file options source=/conf.FDO262921L3 destination=/bootflash/poap_conf.cfg login_timeout=120 destination_tmp=poap_conf.cfg.tmp - script.sh 2026 Sep 22 18:24:17 switch %\$ VDC-1 %\$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Transferring using tftp from /conf.FDO262921L3 to bootflash:/poap_conf.cfg.tmp hostname 192.168.100.1 vrf default - script.sh 2026 Sep 22 18:24:17 switch %\$ VDC-1 %\$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Command is : terminal dont-ask ; terminal password

; copy tftp://root@192.168.100.1/conf.FDO262921L3 bootflash:/poap_conf.cfg.tmp vrf default - script.sh 2026 Sep 22 18:24:18 switch %\$ VDC-1 %\$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - *** Downloaded file is of size 120 *** - script.sh

이러한 메시지는 디바이스별 컨피그레이션이 성공적으로 검색되었고 처리를 위해 로컬에 저장되었음을 확인합니다.

패킷 캡처로 컨피그레이션 검색 확인

패킷 캡처는 Nexus 스위치에 의해 생성된 TFTP 요청에 대한 독립적인 확인을 제공합니다.

검증된 예:

```
sudo tcpdump -nn -r /tmp/poap-run1.pcap 'udp dst port 69' reading from file /tmp/poap-run1.pcap, link-type EN10MB (Ethernet), snapshot length 262144 13:38:19.955108 IP 192.168.100.108.58550 > 192.168.100.1.69: TFTP, length 29, RRQ "conf.FDO262921L3.md5" octet 13:38:20.651491 IP 192.168.100.108.58680 > 192.168.100.1.69: TFTP, length 25, RRQ "conf.FDO262921L3" octet
```

이 절차에서는 스위치가 디바이스 컨피그레이션을 검색하기 전에 컨피그레이션 무결성 파일을 요청했음을 확인합니다.

TFTP 서버 로그에는 두 전송 모두 기록되었습니다.

```
Sep 22 14:26:36 calo-ubuntu24-00505692aeaf dnsmasq-tftp[144902]: sent /srv/tftp/conf.FDO262921L3.md5 to 192.168.100.108
Sep 22 14:26:37 calo-ubuntu24-00505692aeaf dnsmasq-tftp[144902]: sent /srv/tftp/conf.FDO262921L3 to 192.168.100.108
```

컨피그레이션 무결성 확인

구성이 다운로드되면 스크립트는 MD5 체크섬을 계산하고 이를 예상 값과 비교합니다.

이 예에 표시된 프로비저닝 시도는 다음을 생성했습니다.

```
2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Verifying MD5
checksum - script.sh 2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-
MAC[88:FC:5D:BA:D7:7F] - file name is bootflash:poap_conf.cfg - script.sh 2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-
SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - INFO: md5sum a1a51058d79de0348c6cd1934e08acb4 (recalculated) -
script.sh 2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] -
Verifying MD5 checksum of /bootflash/poap_conf.cfg (size 120) - script.sh 2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-
SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - md5given = a1a51058d79de0348c6cd1934e08acb4 md5calculated =
a1a51058d79de0348c6cd1934e08acb4 - script.sh 2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG:
S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - MD5 match for file = /bootflash/poap_conf.cfg - script.sh
```

일치하는 값은 이 프로비저닝 시도 중에 다운로드된 컨피그레이션이 해당 무결성 정보와 일치하는지 확인합니다.

구성 처리 확인

무결성 검사가 성공하면 스크립트는 대상 Cisco NX-OS 이미지에 대해 컨피그레이션을 준비해야 하는 방법을 결정합니다.

이 예에서는 다음을 수행합니다.

```
026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Split Config file -
script.sh 2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Target
image supports bootstrap replay. Split config is not required. - script.sh 2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-
SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Skip split config as it isn't needed with nxos64-cs.10.6.2.F.bin - script.sh
```

이 출력은 대상 이미지가 부트스트랩 재생을 지원하고 이 프로비저닝 시도에 컨피그레이션 분할이 필요하지 않음을 확인합니다.

이 단계에서는 디바이스별 컨피그레이션이 확인, 검색, 검증되고 나머지 POAP 워크플로에 대해 준비되었습니다.

이미지 처리 확인

디바이스 컨피그레이션을 검색하고 검증한 후 POAP 스크립트가 대상 Cisco NX-OS 이미지를 처리하는 방법을 확인합니다.

검증된 구현에서는 nxos64-cs.10.6.2.F.bin을 사용합니다.

스크립트는 먼저 해당 MD5 정보를 검색합니다.

```
2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Starting Copy of MD5. src=/nxos64-cs.10.6.2.F.bin.md5 dest = /bootflash/nxos64-cs.10.6.2.F.bin.md5 - script.sh 2026 Sep 22 18:24:18 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - File transfer_protocol = tftp - script.sh
```

그런 다음 MD5 파일은 bootflash에 저장됩니다.

```
2026 Sep 22 18:24:19 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - *** Downloaded file is of size 57 *** - script.sh 2026 Sep 22 18:24:19 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Renamed /bootflash/nxos64-cs.10.6.2.F.bin.md5.tmp to /bootflash/nxos64-cs.10.6.2.F.bin.md5 - script.sh
```

스크립트는 프로비저닝 서버에서 받은 체크섬을 읽습니다.

```
2026 Sep 22 18:24:19 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - MD5 for system image from server: ce50c3f0e387bee578e615a373a4ede0 - script.sh
```

그런 다음 bootflash에 이미 있는 이미지의 체크섬을 계산합니다.

```
2026 Sep 22 18:24:19 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Verifying MD5 checksum - script.sh 2026 Sep 22 18:24:19 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - file name is bootflash:nxos64-cs.10.6.2.F.bin - script.sh 2026 Sep 22 18:24:30 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - INFO: md5sum ce50c3f0e387bee578e615a373a4ede0 (recalculated) - script.sh 2026 Sep 22 18:24:30 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Verifying MD5 checksum of /bootflash/nxos64-cs.10.6.2.F.bin (size 2952914432) - script.sh 2026 Sep 22 18:24:30 switch %$ VDC-1 %$
```

```
%USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - md5given = ce50c3f0e387bee578e615a373a4ede0
md5calculated = ce50c3f0e387bee578e615a373a4ede0 - script.sh
```

스크립트는 기존 이미지가 예상 이미지와 일치하는지 확인합니다.

```
2026 Sep 22 18:24:30 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - MD5 match for
file = /bootflash/nxos64-cs.10.6.2.F.bin - script.sh 2026 Sep 22 18:24:30 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG:
S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - File /bootflash/nxos64-cs.10.6.2.F.bin already exists and MD5 matches - script.sh
```

일치하는 값을 통해 필요한 Cisco NX-OS 이미지가 bootflash에 이미 있음을 확인할 수 있습니다. 무결성 검사를 통과했습니다. 따라서 이미지의 다른 복사본을 다운로드할 필요가 없습니다.

이미지 MD5 검색 확인

패킷 캡처는 또한 Nexus가 이미지 무결성 파일을 요청했음을 확인합니다.

```
sudo tcpdump -nn -r /tmp/poap-run1.pcap 'udp dst port 69' reading from file /tmp/poap-run1.pcap, link-type EN10MB (Ethernet), snapshot
length 262144 13:38:21.737101 IP 192.168.100.108.58866 > 192.168.100.1.69: TFTP, length 35, RRQ "nxos64-cs.10.6.2.F.bin.md5" octet
```

TFTP 서버 로그에 동일한 전송이 기록되었습니다.

```
Sep 22 14:26:38 calo-ubuntu24-00505692aeaf dnsmasq-tftp[144902]: sent /srv/tftp/nxos64-cs.10.6.2.F.bin.md5 to 192.168.100.108
```

이러한 출력은 프로비저닝 서버에서 nxos64-cs.10.6.2.F.bin.md5를 검색하는 것을 확인합니다.

소프트웨어 설치 워크플로 확인

이미지 파일을 다운로드할 필요는 없지만, 참조 스크립트는 Cisco NX-OS 설치 워크플로를 계속 진행했습니다.

검증된 예는 다음과 같이 생성됩니다.

```
2026 Sep 22 18:24:31 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-MAC[88:FC:5D:BA:D7:7F] - Done copying the
second scheduled cfg - script.sh 2026 Sep 22 18:24:31 switch %$ VDC-1 %$ %USER-1-SYSTEM_MSG: S/N[FDO262921L3]-
```

MAC[88:FC:5D:BA:D7:7F] - terminal dont-ask ; install all nxos bootflash:/nxos64-cs.10.6.2.F.bin no-reload non-interruptive - script.sh

기존 이미지가 무결성 검사를 통과했으며 다른 파일 전송이 필요하지 않았습니다. 그러나 참조 스크립트는 여전히 검증된 이미지에 대해 소프트웨어 설치 프로세스를 호출했습니다.



참고: 기존 이미지와 일치하는 MD5는 불필요한 이미지 다운로드를 방지할 수 있습니다. 따라서 나머지 소프트웨어 설치 워크플로를 건너뛸 필요는 없습니다.

소프트웨어 처리가 완료되면 POAP는 나머지 프로비저닝 작업을 계속하고 스위치가 다시 로드될 수 있도록 준비합니다.

구성 재생 확인

POAP 스크립트가 성공적으로 완료되면 스위치가 다시 로드 및 컨피그레이션 재생 프로세스를 진행하는지 확인합니다.

검증된 예에서는 성공적인 스크립트 실행 후 다시 로드를 보고했습니다.

```
2026 Sep 22 18:31:08 switch %$ VDC-1 %$ %POAP-2-POAP_SCRIPT_EXEC_SUCCESS: [FDO262921L3-88:FC:5D:BA:D7:7F] - POAP
script execution success 2026 Sep 22 18:31:09 switch %$ VDC-1 %$ %POAP-2-POAP_RELOAD_DEVICE: [FDO262921L3-
88:FC:5D:BA:D7:7F] - Reload device
```

이러한 메시지는 스크립트 실행 단계가 완료되었고 설치 후 다시 로드 단계로 전환되었음을 확인합니다.

스위치가 다시 시작되면 Cisco NX-OS에서 POAP 재생 파일을 탐지합니다.

```
2026 Sep 22 18:33:49 %$ VDC-1 %$ %DAEMON-2-SYSTEM_MSG: <<%ASCII-CFG-2-CONF_CONTROL>> Poap replay
/bootflash/poap_replay01.cfg - ascii_cfg_server[17165]
```

그런 다음 부트스트랩 재생이 시작되고 완료되었습니다.

```
2026 Sep 22 18:34:26 %$ VDC-1 %$ %ASCII-CFG-2-CONFIG_REPLAY_STATUS: Bootstrap Replay Started. 2026 Sep 22 18:34:27 %$
VDC-1 %$ %ASCII-CFG-2-CONFIG_REPLAY_STATUS: Bootstrap Replay Done.
```

이러한 이벤트를 통해 Cisco NX-OS가 POAP 중에 준비된 컨피그레이션 처리를 시작했음을 확인할 수 있습니다.

ASCII 컨피그레이션 재생 확인

부트스트랩 재생이 완료되면 NX-OS는 프로비저닝된 컨피그레이션을 처리합니다.

제어된 유효성 검사 시도로 다음 메시지가 생성되었습니다.

```
2026 Sep 22 18:35:42 switch %$ VDC-1 %$ %ASCII-CFG-2-CONFIG_REPLAY_STATUS: Ascii Replay Started. 2026 Sep 22 18:35:44
N9K-POAP-SW1 %$ VDC-1 %$ %ASCII-CFG-2-CONFIG_REPLAY_STATUS: Ascii Replay Done. 2026 Sep 22 20:03:26 switch
%VDC_MGR-5-VDC_HOSTNAME_CHANGE: vdc 1 hostname changed to N9K-POAP-SW1 2026 Sep 22 20:03:27 N9K-POAP-SW1
%ETHPORT-5-IF_UP: Interface loopback100 is up 2026 Sep 22 20:03:28 N9K-POAP-SW1 %ASCII-CFG-2-CONFIG_REPLAY_STATUS:
Ascii Replay Done. 2026 Sep 22 20:03:29 N9K-POAP-SW1 %ASCII-CFG-2-CONF_CONTROL: System ready
```

이 메시지는 ASCII 컨피그레이션 재생을 표시하고 프로비저닝된 컨피그레이션이 적용되었음을 직접 입증합니다.

호스트 이름이 N9K-POAP-SW1로 변경되고 루프백100이 프로비저닝된 컨피그레이션의 일부로 작동하게 되었습니다.

POAP 완료 확인

컨피그레이션 재생이 완료되면 Cisco NX-OS는 결과 컨피그레이션을 저장합니다.

```
2026 Sep 22 19:24:03 N9K-POAP-SW1 %$ VDC-1 %$ %ASCII-CFG-2-CONF_CONTROL: System ready
[#####] 100% Copy complete, now saving to disk (please wait)... Copy complete. Auto provisioning
complete
```

자동 프로비저닝 완료 메시지는 POAP 워크플로의 완료를 확인합니다.

이 단계에서는 실행 중인 컨피그레이션 및 시작 컨피그레이션을 확인하여 의도한 Day-0 컨피그레이션이 활성화 및 지속적인지 확인합니다.

프로비저닝된 컨피그레이션 확인

POAP가 완료되면 의도한 Day-0 컨피그레이션이 활성화 상태이고 지속적인지 확인합니다. 프로비저닝된 호스트 이름, 관리 계정, 인터페이스 컨피그레이션, Cisco NX-OS 이미지 및 시스템 전반의 POAP 상태를 확인합니다.

```
N9K-POAP-SW1# show hostname N9K-POAP-SW1 N9K-POAP-SW1# show user-account admin user:admin this user account has no expiry date roles:network-admin N9K-POAP-SW1# show ip interface brief IP Interface Status for VRF "default"(1) Interface IP Address Interface Status Lo100 10.255.255.100 protocol-up/link-up/admin-up N9K-POAP-SW1# show running-config interface loopback100 !Command: show running-config interface loopback100 !Running configuration last done at: Tue Sep 22 20:03:48 2026 !Time: Tue Sep 22 23:17:52 2026 version 10.6(2) Bios:version 05.53 interface loopback100 description POAP-VALIDATION ip address 10.255.255.100/32 N9K-POAP-SW1# show startup-config interface loopback100 !Command: show startup-config interface loopback100 !Time: Tue Sep 22 23:17:57 2026 !Startup config saved at: Tue Sep 22 20:03:48 2026 version 10.6(2) Bios:version 05.53 interface loopback100 description POAP-VALIDATION ip address 10.255.255.100/32 N9K-POAP-SW1# show startup-config | include username username admin password 5

role network-admin N9K-POAP-SW1# show version | include NXOS NXOS: version 10.6(2) [Feature Release] Host NXOS: version 10.6(2) NXOS image file is: bootflash:///nxos64-cs.10.6.2.F.bin NXOS compile time: 11/30/2025 12:00:00 [12/12/2025 12:10:23] NXOS boot mode: LXC N9K-POAP-SW1# show system poap System-wide POAP is enabled
```

이 출력에서는 디바이스별 컨피그레이션이 성공적으로 적용되었고 시작 컨피그레이션에 저장되었음을 확인합니다.

또한 스위치에서 예상 Cisco NX-OS 이미지를 실행 중이며 시스템 전반의 POAP가 활성화된 상태임을 확인합니다.

결론

이 문서에서는 DHCP 검색, 스크립트 검색 및 실행, 디바이스별 컨피그레이션 선택, 파일 무결성 검증, Cisco NX-OS 이미지 처리, 컨피그레이션 재생을 비롯한 주요 POAP 구성 요소 및 워크플로에 대해 설명합니다.

검증된 구현에서는 Cisco Nexus 9000 스위치가 필요한 프로비저닝 리소스를 확보하고 Day-0 컨피그레이션을 적용하고 그 결과 컨피그레이션을 유지하는 방법을 보여줍니다.

확인 프로세스에서는 또한 프로비저닝 워크플로의 각 주요 단계를 확인하기 위해 콘솔 메시지, POAP 로그, 서버 로그 및 패킷 캡처를 상호 연결하는 방법을 보여줍니다.

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.