OpenAPI를 사용하여 ISE 3.3에서 ISE 인증서 정보 검색

목차

소개

이 문서에서는 openAPI를 사용하여 Cisco ISE(Identity Services Engine) 인증서를 관리하는 절차에 대해 설명합니다.

배경

엔터프라이즈 네트워크 보안 및 관리의 복잡성이 증가하는 상황에서 Cisco ISE 3.1은 인증서 라이프사이클 관리를 간소화하는 OpenAPI 형식의 API를 도입하여 효율적이고 안전한 인증서 운영을 위한 표준화되고 자동화된 인터페이스를 제공함으로써 관리자가 강력한 보안 방식을 적용하고 네트워크 컴플라이언스를 유지할 수 있도록 지원합니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- Cisco ISE(Identity Services Engine)
- REST API
- 비단뱀

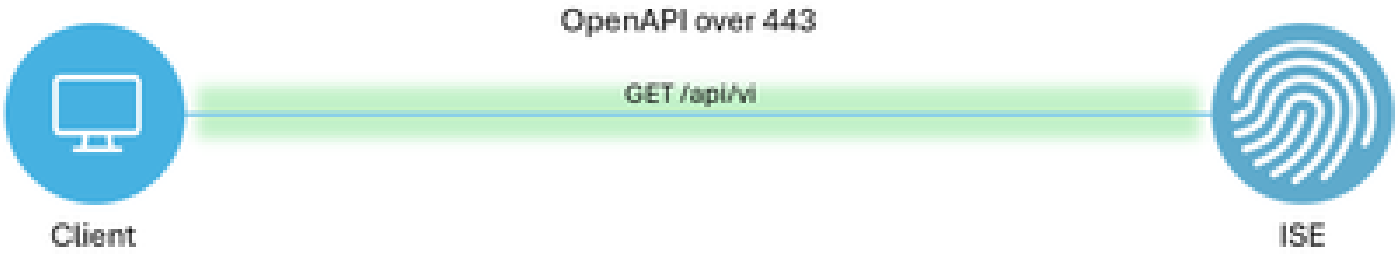
사용되는 구성 요소

- ISE 3.3
- 파이썬 3.10.0

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.

구성

네트워크 다이어그램

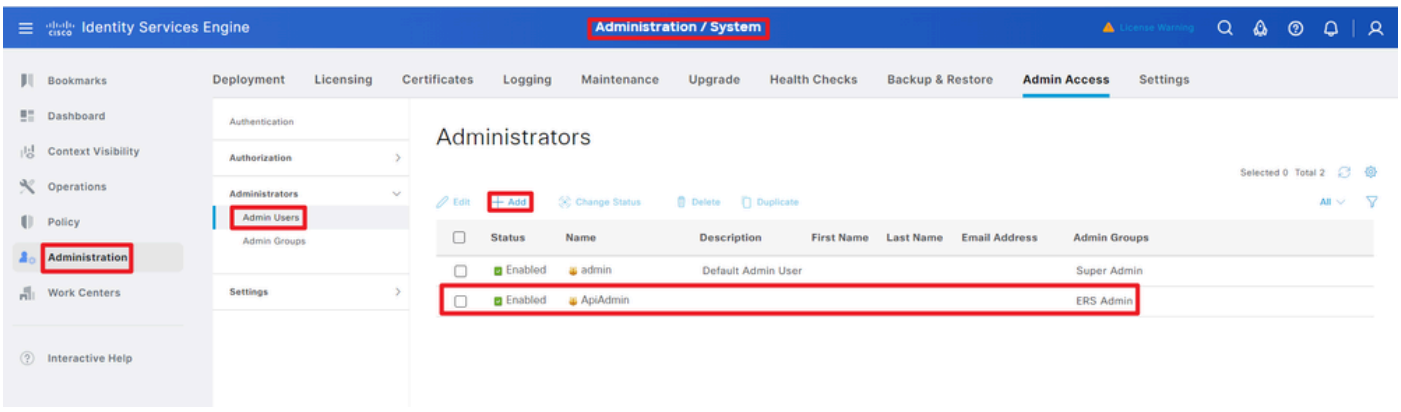


토폴로지

ISE의 컨피그레이션

1단계: Open API 관리자 계정 추가

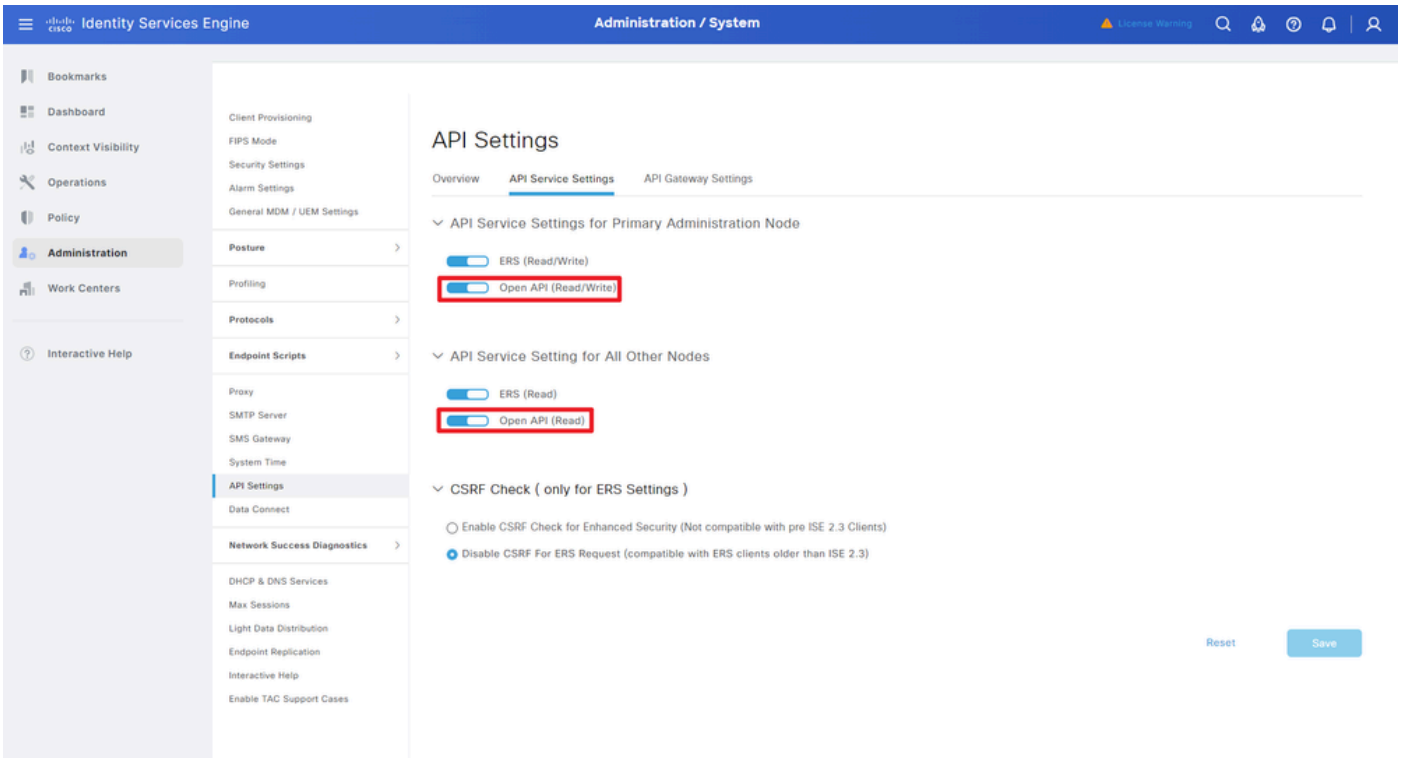
API 관리자를 추가하려면 Administration -> System -> Administration -> Administrators -> Admin Users -> Add로 이동합니다.



API 관리자

2단계: ISE에서 Open API 활성화

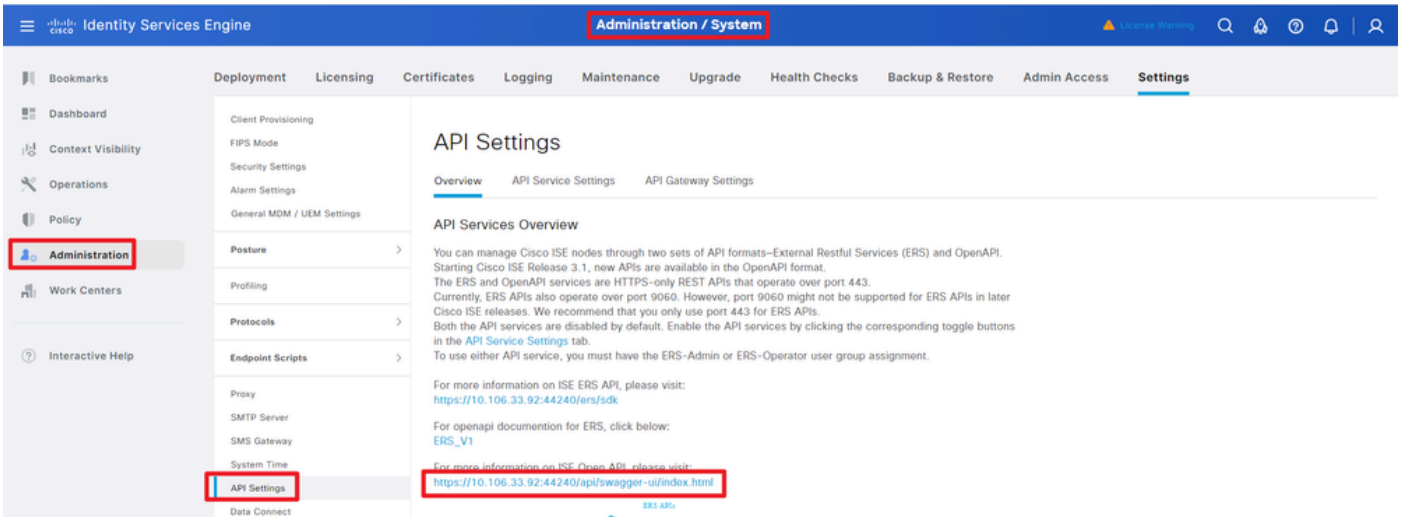
Open API는 ISE에서 기본적으로 비활성화되어 있습니다. 활성화하려면 Administration(관리) > System(시스템) > API Settings(API 설정) > API Service Settings(API 서비스 설정)로 이동합니다. Open API 옵션을 전환합니다. 저장을 클릭합니다.



OpenAPI 활성화

3단계: ISE 개방형 API 탐색

administration(관리) > System(시스템) > API Settings(API 설정) > Overview(개요)로 이동합니다. API 방문 열기 링크를 클릭합니다.



OpenAPI 방문

Python 예

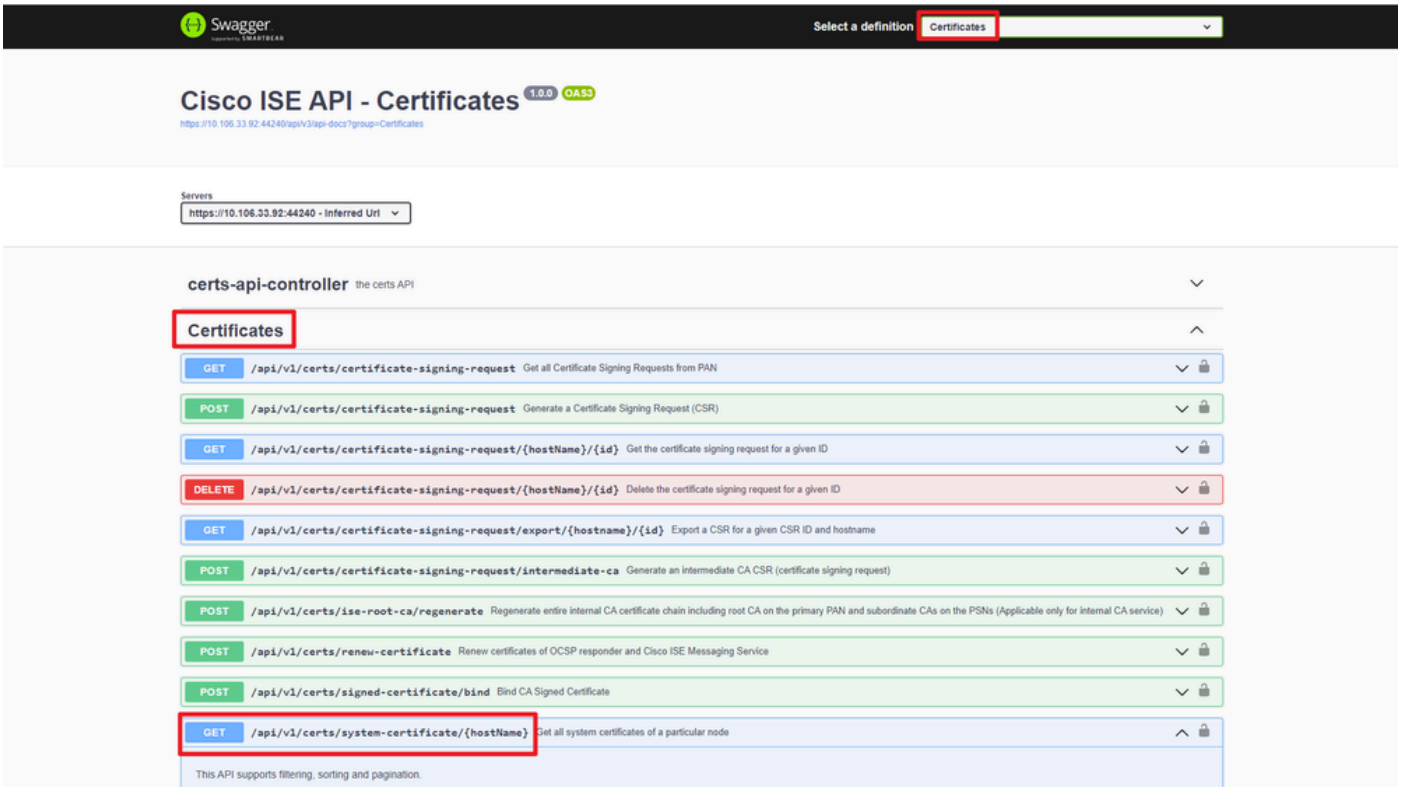
특정 노드의 모든 시스템 인증서 가져오기

API는 특정 ISE 노드의 모든 인증서를 나열합니다.

1단계: API 호출에 필요한 정보입니다.

방법	가져오기
URL	https://<ISE-PAN-IP>/api/v1/certs/system-certificate/<ISE-Node-Hostname>
자격 증명	Open API 계정 자격 증명 사용
헤더	수락: application/json Content-Type: application/json

2단계: 특정 ISE 노드의 인증서를 검색하는 데 사용되는 URL을 찾습니다.



API URI

3단계: Python Code의 예를 소개합니다. 내용을 복사하여 붙여넣습니다. ISE IP, 사용자 이름, 비밀번호를 교체합니다. 실행할 python 파일로 저장합니다.

ISE와 python 코드 예제를 실행하는 디바이스 간의 올바른 연결을 확인합니다.

<#root>

```
from requests.auth import HTTPBasicAuth
import requests

requests.packages.urllib3.disable_warnings()

if __name__ == "__main__":

    url = "
https://10.106.33.92/api/v1/certs/system-certificate/ISE-DLC-CFME02-PSN
"
```

```
    headers = {
"Accept": "application/json", "Content-Type": "application/json"
}
    basicAuth = HTTPBasicAuth(
"ApiAdmin", "Admin123"
)
    response = requests.get(url=url, auth=basicAuth, headers=headers, verify=False)
    print("Return Code:")
    print(response.status_code)
    print("Expected Outputs:")
    print(response.json())
```

다음은 예상 출력의 예입니다.

Return Code:
200
Expected Outputs:
{'response': [{'id': '5b5b28e4-2a51-495c-8413-610190e1070b', 'friendlyName': 'Default self-signed saml server certificate - CN=SAML_ISE-DLC-CFME0

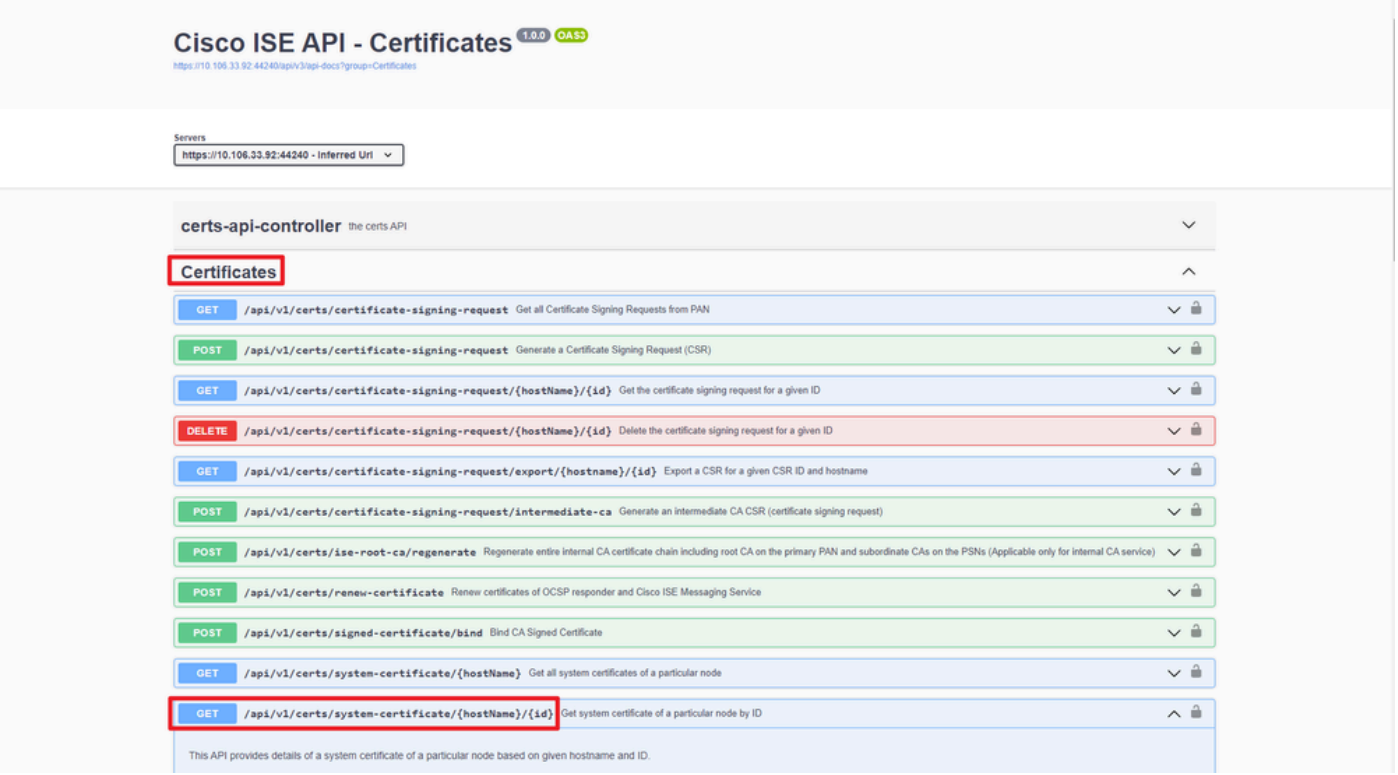
ID로 특정 노드의 시스템 인증서 가져오기

이 API는 제공된 호스트 이름 및 ID를 기반으로 특정 노드의 시스템 인증서에 대한 세부 정보를 제공합니다.

1단계: API 호출에 필요한 정보입니다.

방법	가져오기
URL	https://<ISE-PAN-IP>/api/v1/certs/system-certificate/<ISE-Node-Hostname>/<ID-Of-Certificate>
자격 증명	Open API 계정 자격 증명 사용
헤더	수락: application/json Content-Type: application/json

2단계: 지정된 호스트 이름 및 ID를 기반으로 특정 노드의 인증서를 검색하는 데 사용되는 URL을 찾습니다.



API URI

3단계: Python Code의 예입니다. 내용을 복사하여 붙여넣습니다. ISE IP, 사용자 이름, 비밀번호를 교체합니다. 실행할 python 파일로 저장합니다.

ISE와 python 코드 예제를 실행하는 디바이스 간의 올바른 연결을 확인합니다.

<#root>

from requests.auth import HTTPBasicAuth import requests requests.packages.urllib3.disable_warnings() if __name__ == "__main__": url = "

https://10.106.33.92/api/v1/certs/system-certificate/ISE-DLC-CFME02-PSN/5b5b28e4-2a51-495c-8413-610190e1

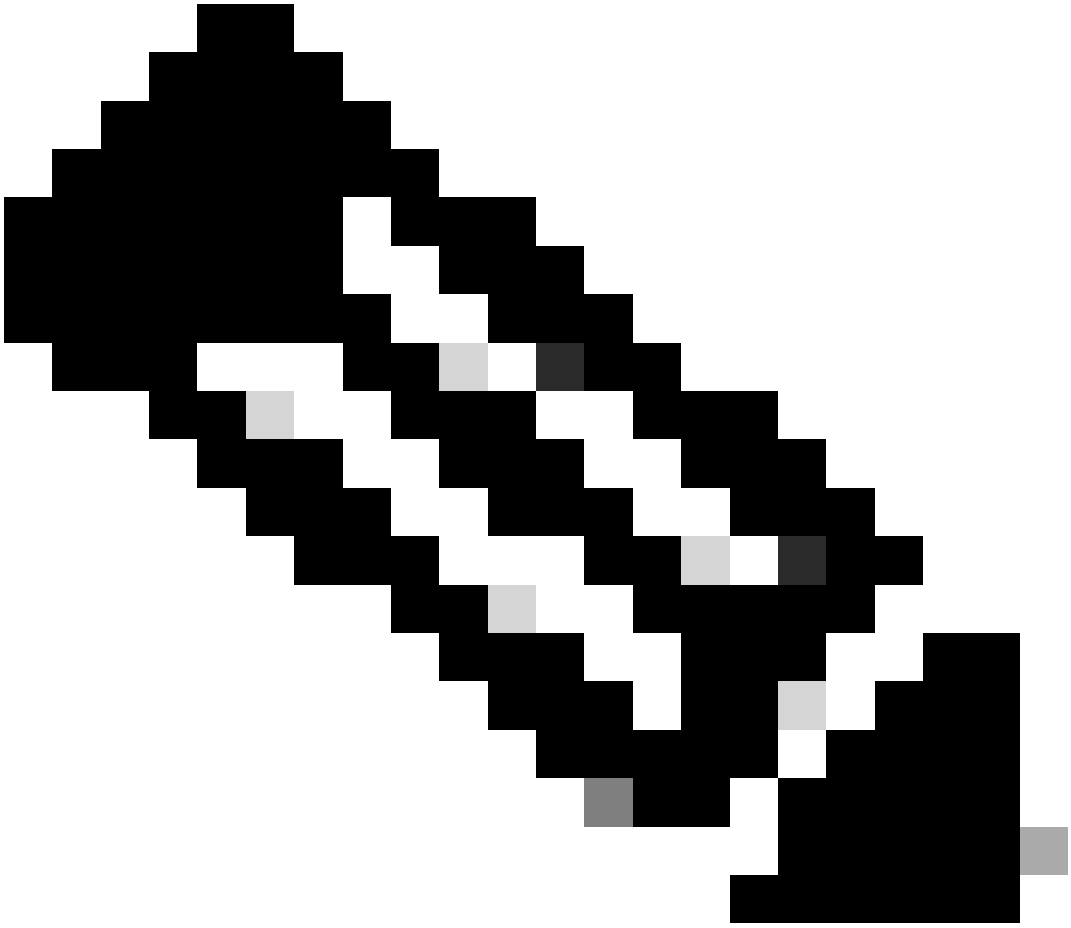
" headers = {

"Accept": "application/json", "Content-Type": "application/json"

} basicAuth = HTTPBasicAuth(

"ApiAdmin", "Admin123"

) response = requests.get(url=url, auth=basicAuth, headers=headers, verify=False) print("Return Code:")



참고: ID는 "특정 노드의 모든 시스템 인증서 가져오기"의 3단계에서 API 출력에서 가져온 것입니다. 예를 들어, 5b5b28e4-2a51-495c-8413-610190e1070b는 "Default self-signed saml server certificate - CN=SAML_ISE-DLC-CFME02-PSN.cisco.com"입니다.

다음은 예상 출력의 예입니다.

Return Code:
200
Expected Outputs:
{'response': {'id': '5b5b28e4-2a51-495c-8413-610190e1070b', 'friendlyName': 'Default self-signed saml server certificate - CN=SAML_ISE-DLC-CFME02

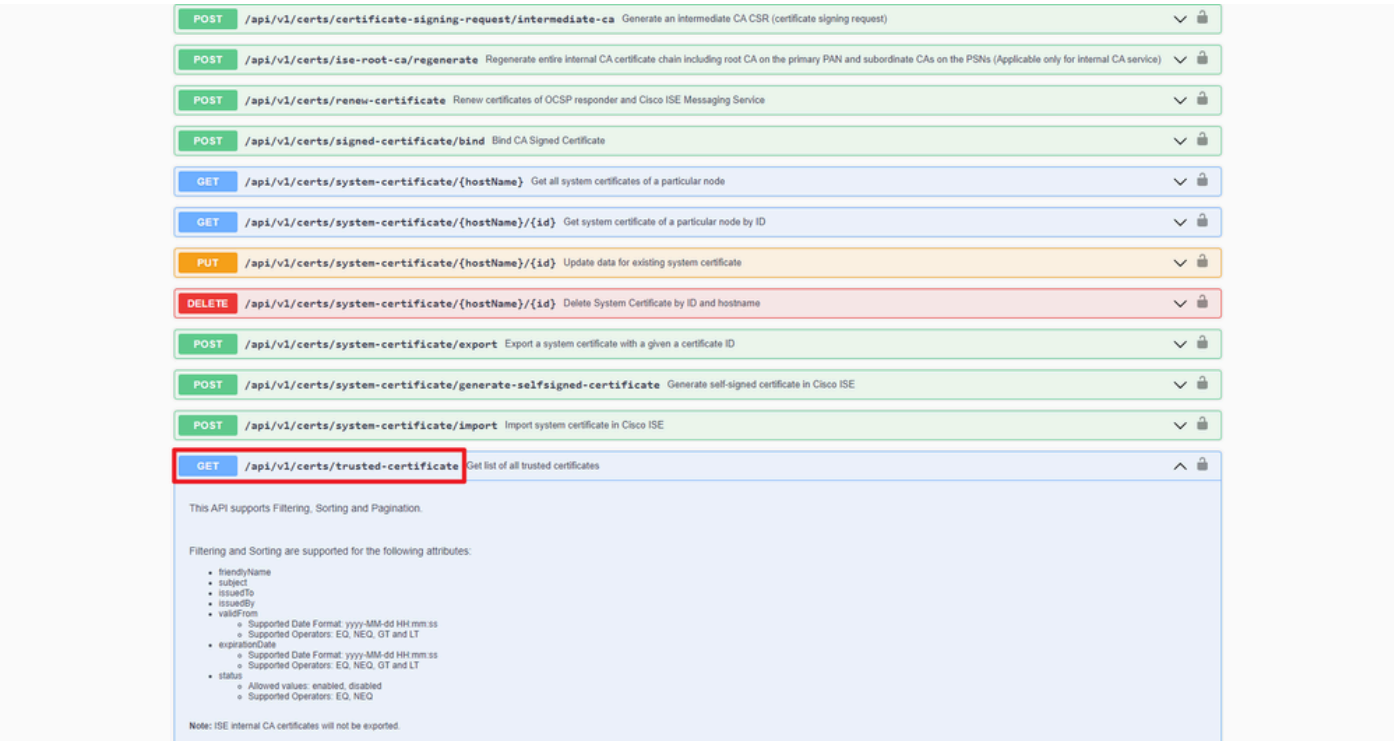
신뢰할 수 있는 모든 인증서 목록 가져오기

API는 ISE 클러스터의 모든 신뢰할 수 있는 인증서를 나열합니다.

1단계: API 호출에 필요한 정보입니다.

방법	가져오기
URL	https://<ISE-PAN-IP>/api/v1/certs/trusted-certificate
자격 증명	Open API 계정 자격 증명 사용
헤더	수락: application/json Content-Type: application/json

2단계: 신뢰할 수 있는 인증서를 검색하는 데 사용되는 URL을 찾습니다.



API URI

3단계: Python Code의 예입니다. 내용을 복사하여 붙여넣습니다. ISE IP, 사용자 이름, 비밀번호를 교체합니다. 실행할 python 파일로 저장합니다.

ISE와 python 코드 예제를 실행하는 디바이스 간의 올바른 연결을 확인합니다.

<#root>

from requests.auth import HTTPBasicAuth import requests requests.packages.urllib3.disable_warnings() if __name__ == "__main__": url = "

https://10.106.33.92/api/v1/certs/trusted-certificate

" headers = {

"Accept": "application/json", "Content-Type": "application/json"

} basicAuth = HTTPBasicAuth(

"ApiAdmin", "Admin123"

```
) response = requests.get(url=url, auth=basicAuth, headers=headers, verify=False) print("Return Code:")
```

다음은 예상 출력의 예입니다. (생략)

```
Return Code:
200
Expected Outputs:
{'response': [{'id': '147d97cc-6ce9-43d7-9928-8cd0fa83e140', 'friendlyName': 'VeriSign Class 3 Public Primary Certification Authority', 'subject': 'CN=Ver
```

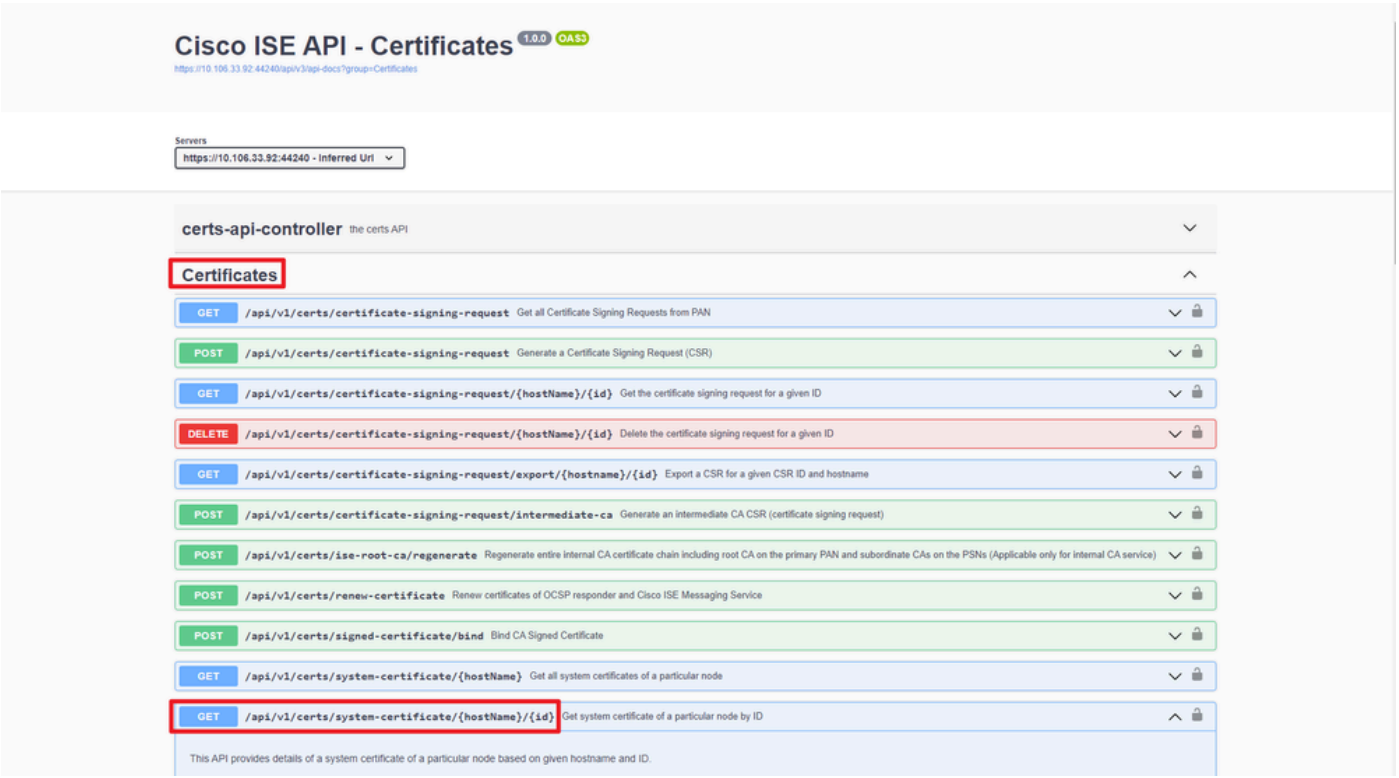
ID로 트러스트 인증서 가져오기

이 API는 지정된 ID를 기반으로 신뢰 인증서의 세부 정보를 표시할 수 있습니다.

1단계: API 호출에 필요한 정보입니다.

방법	가져오기
URL	https://<ISE-PAN-IP>/api/v1/certs/trusted-certificate/<ID-Of-Certificate>
자격 증명	Open API 계정 자격 증명 사용
헤더	수락: application/json Content-Type: application/json

2단계: 구축 정보를 검색하는 데 사용되는 URL을 찾습니다.



API URI

3단계: Python Code의 예입니다. 내용을 복사하여 붙여넣습니다. ISE IP, 사용자 이름, 비밀번호를 교체합니다. 실행할 python 파일로 저장합니다.

ISE와 python 코드 예제를 실행하는 디바이스 간의 올바른 연결을 확인합니다.

<#root>

from requests.auth import HTTPBasicAuth import requests requests.packages.urllib3.disable_warnings() if __name__ == "__main__": url = "

https://10.106.33.92/api/v1/certs/trusted-certificate/147d97cc-6ce9-43d7-9928-8cd0fa83e140

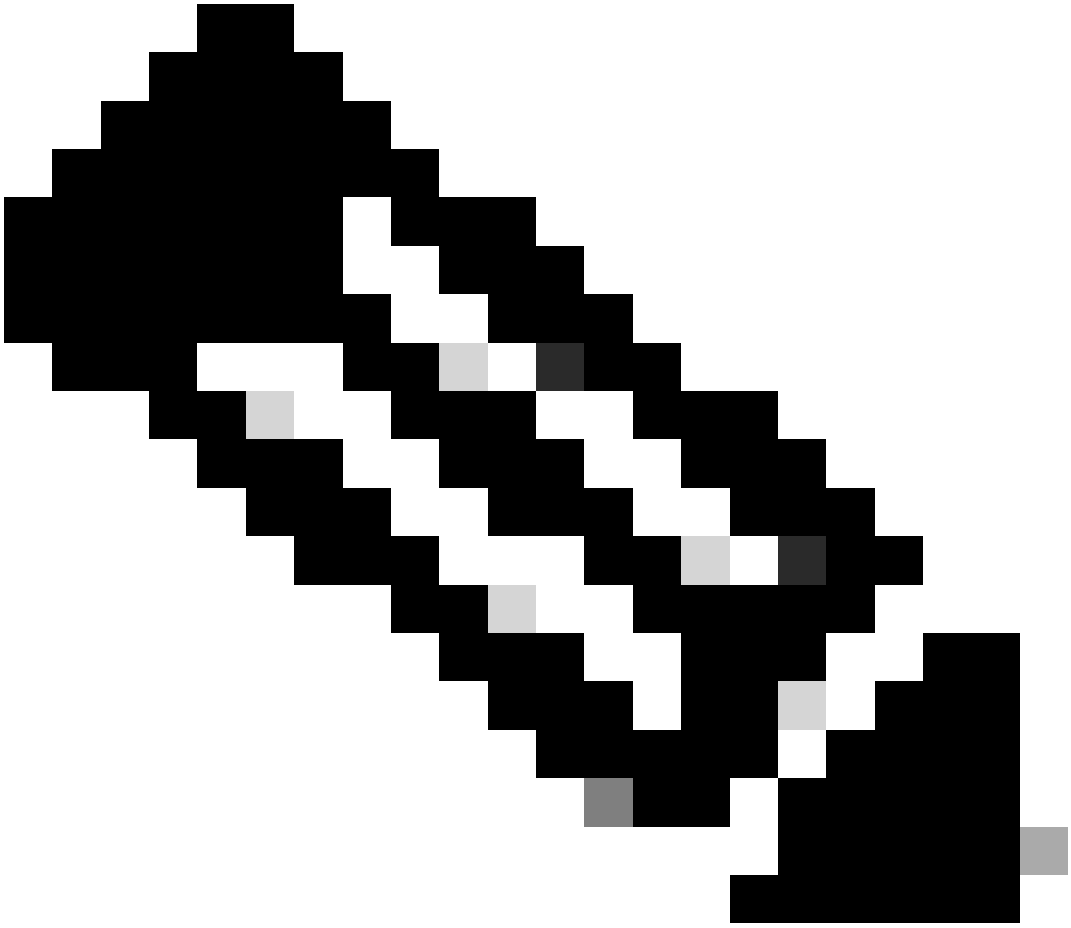
" headers = {

"Accept": "application/json", "Content-Type": "application/json"

} basicAuth = HTTPBasicAuth(

"ApiAdmin", "Admin123"

) response = requests.get(url=url, auth=basicAuth, headers=headers, verify=False) print("Return Code:")



참고: ID는 "모든 신뢰할 수 있는 인증서 목록 가져오기"의 3단계에서 API 출력에서 가져온 것입니다. 예를 들어 147d97cc-6ce9-43d7-9928-8cd0fa83e140은 "VeriSign Class 3 Public Primary Certification Authority"입니다.

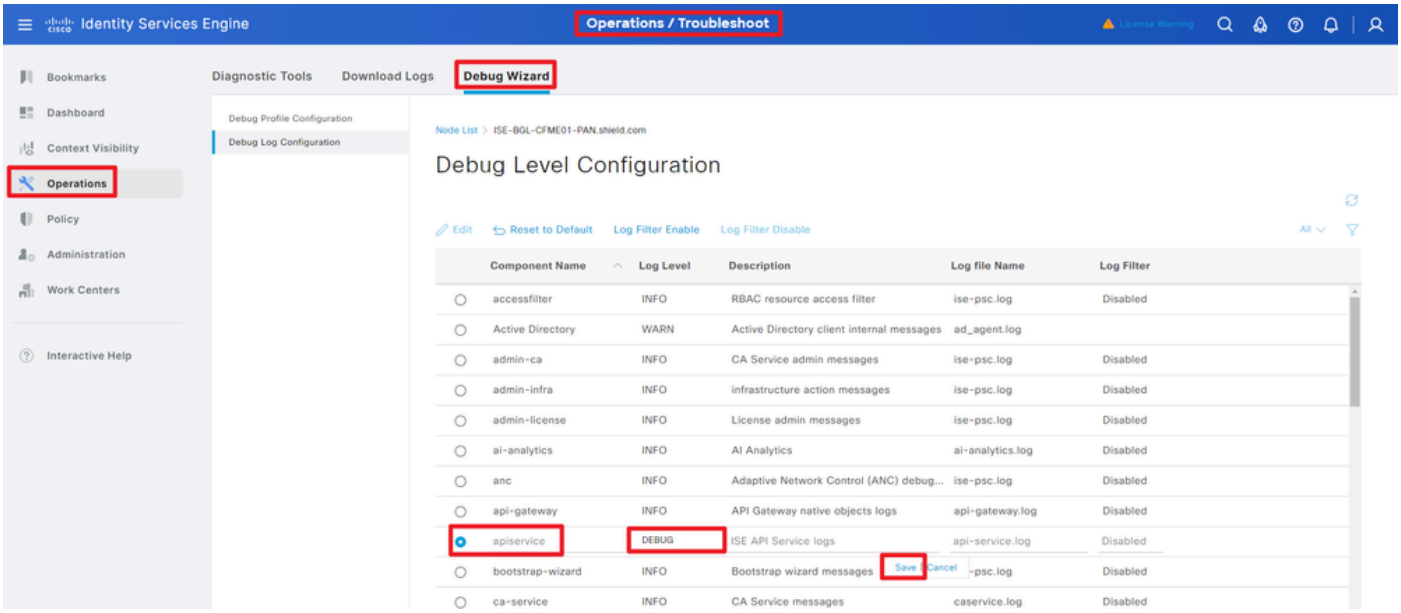
다음은 예상 출력의 예입니다.

Return Code: 200 Expected Outputs: {'response': {'id': '147d97cc-6ce9-43d7-9928-8cd0fa83e140', 'friendlyName': 'VeriSign Class 3 Public Primary Certifi

문제 해결

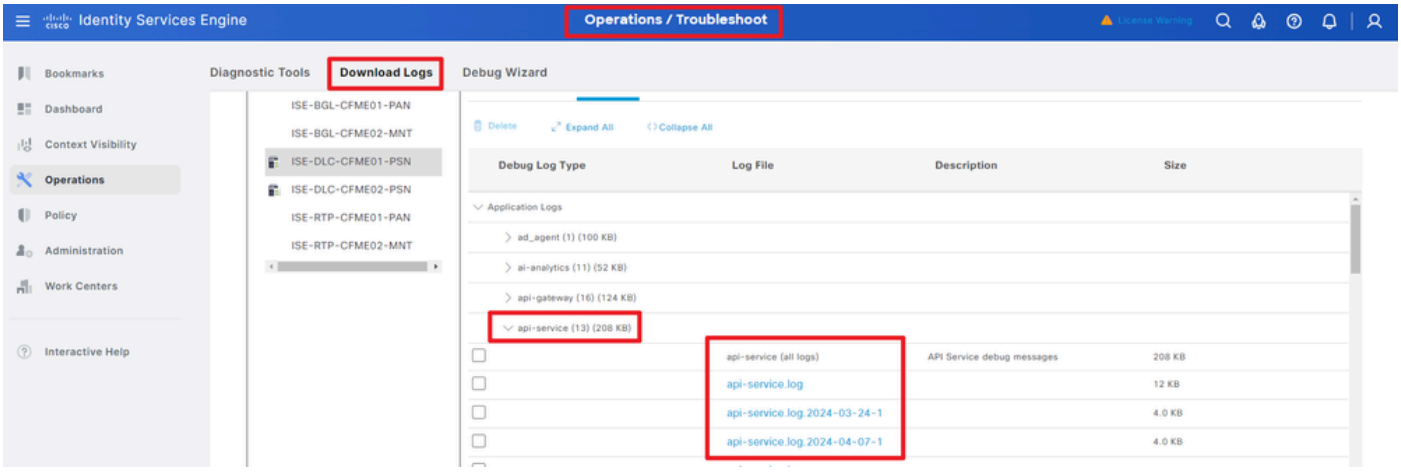
Open API와 관련된 문제를 해결하려면 Debug Log Configuration(디버그 **로그 컨피그레이션) 창**에서 Log Levelfor theapiservicecomponent를 **DEBUG로** 설정합니다.

디버그를 활성화하려면 **Operations(운영) -> Troubleshoot(문제 해결) -> Debug Wizard(디버그 마법사) -> Debug Log Configuration(디버그 로그 컨피그레이션) -> ISE Node(ISE 노드) -> apiservice로 이동합니다.**



API 서비스 디버그

디버그 로그를 다운로드하려면 **Operations(운영) -> Troubleshoot(문제 해결) -> Download Logs(로그 다운로드) -> ISE PAN Node(ISE PAN 노드) -> Debug Logs(디버그 로그)로 이동합니다.**



디버그 로그 다운로드

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.