OpenAPI를 사용하여 ISE 3.3에서 ISE 정책 정보 검색

목차

소개

이 문서에서는 OpenAPI를 사용하여 관리하는 절차에 대해 설명합니다 Cisco ISE(Identity Services Engine) 정책.

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 디바

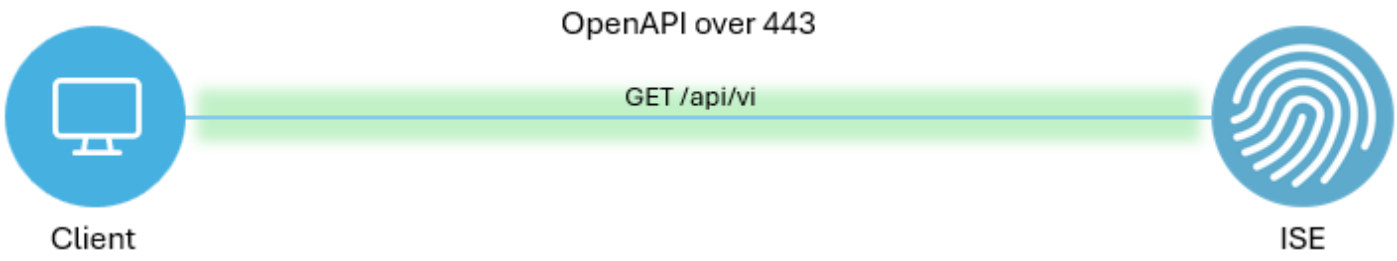
이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

배경 정보

Cisco ISE 3.1 이상에서는 OpenAPI 형식으로 새로운 API를 사용할 수 있습니다. 관리 정책은 상호 운용성을 높이고, 자동화 효율성을 개선하며, 보안을 강화하고, 혁신을 촉진하고, 비용을 절감하여 네트워크 보안 및 관리를 최적화합니다. 이 정책을 통해 ISE는 다른 시스템과 원활하게 통합되고, 자동화된 구성 및 관리를 달성하고, 세분화된 액세스 제어를 제공하고, 서드파티 혁신을 장려하고, 관리 프로세스를 간소화하여 유지 관리 비용을 줄이고 전반적인 ROI를 높일 수 있습니다.

구성

네트워크 다이어그램

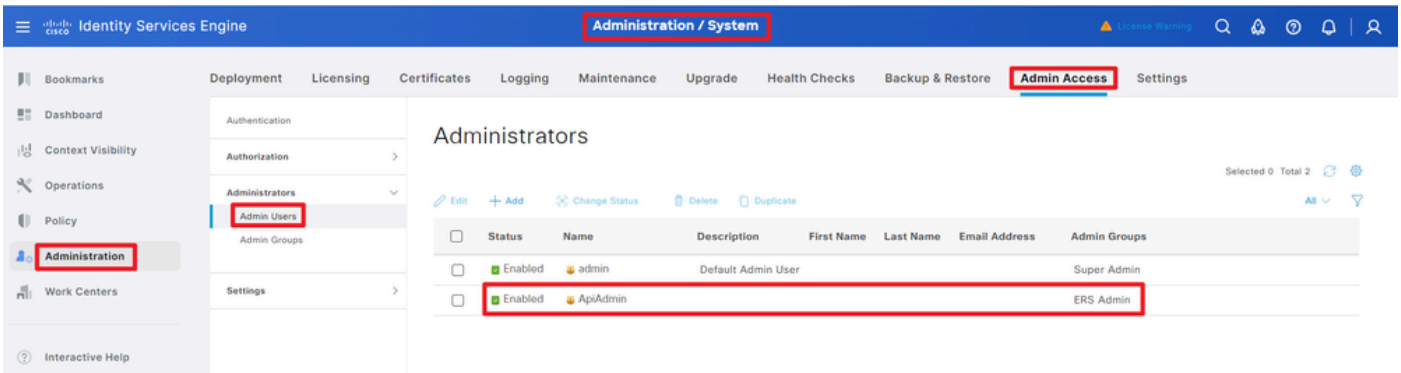


토폴로지

ISE의 컨피그레이션

1단계. OpenAPI 관리자 계정을 추가합니다.

API 관리자를 추가하려면 Administration(관리) > System(시스템) > Admin Access(관리자 액세스) > Administrators(관리자) > Admin Users(관리자 사용자) > Add(추가)로 이동합니다.

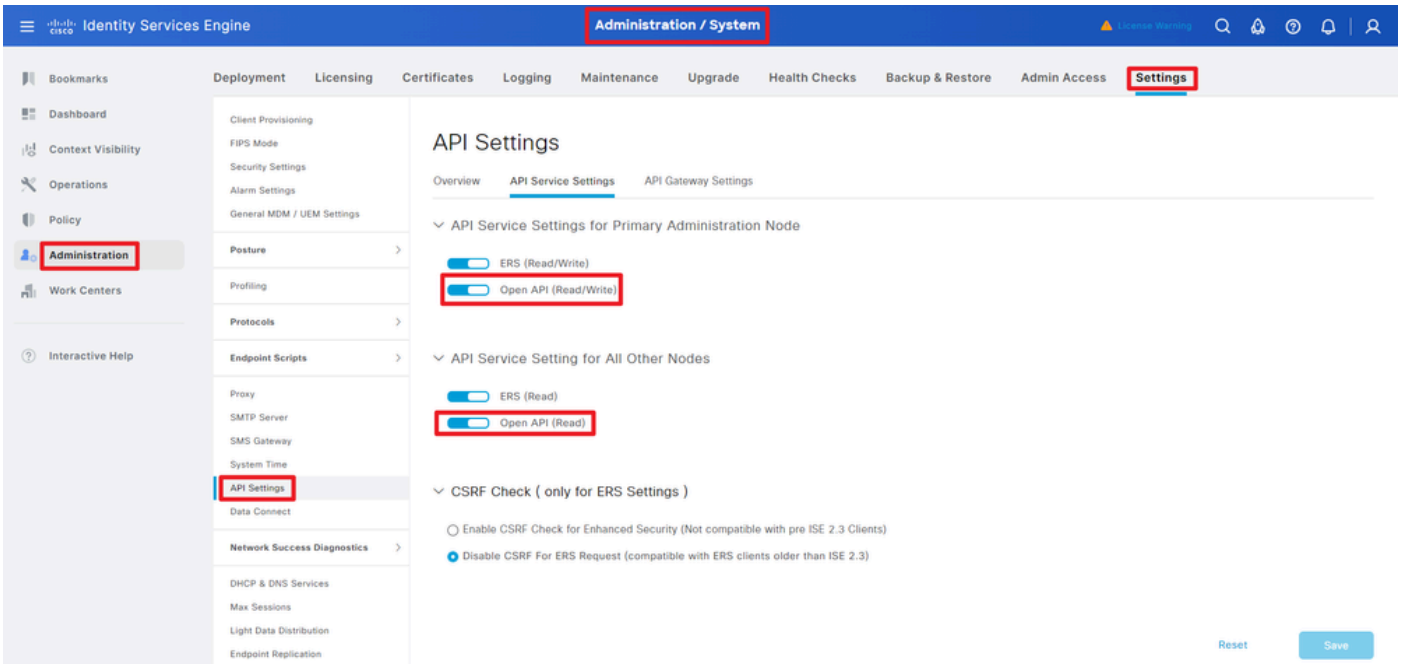


API 관리자

2단계. ISE에서 OpenAPI를 활성화합니다.

Open API는 ISE에서 기본적으로 비활성화되어 있습니다. 활성화하려면 Administration(관리) > System(시스템) > Settings(설정) > API Settings(API 설정) > API Service Settings(API 서비스 설정

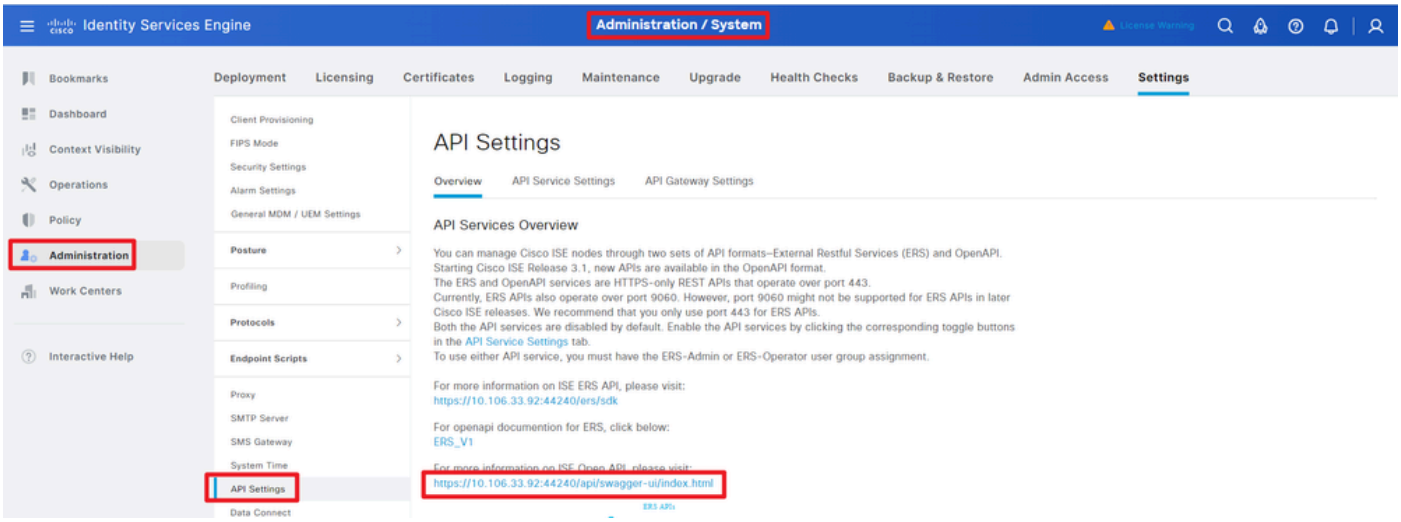
). OpenAPI 옵션을 전환합니다. 클릭 저장.



OpenAPI 활성화

3단계. ISE OpenAPI를 탐색합니다.

로 이동합니다 관리 > 시스템 > 설정 > API 설정 > 개요. 링크를 방문하려면 OpenAPI를 클릭합니다 .



OpenAPI 방문

Python 예

장치 관리자 - 정책 집합 목록

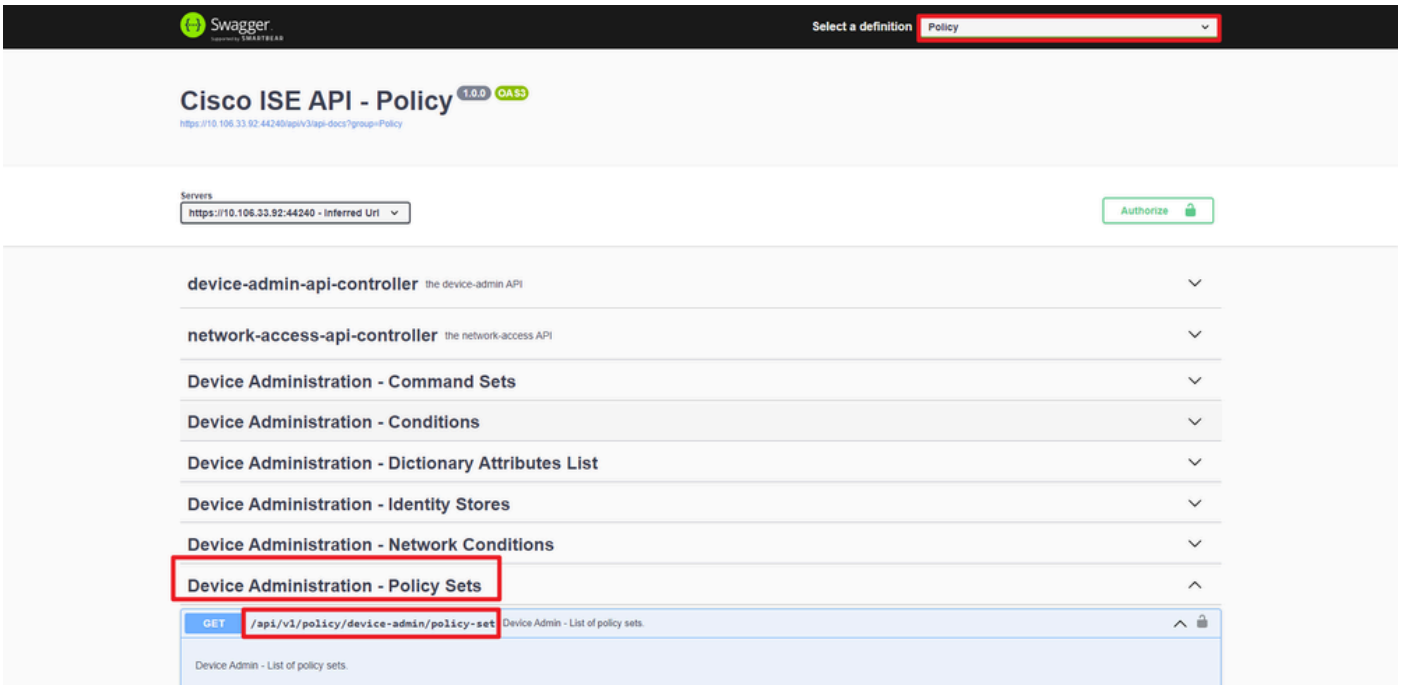
이 API는 디바이스 관리 정책 집합 정보를 검색합니다.

1단계. API 호출에 필요한 정보입니다.

방법	가져오기

URL	https://<ISE-PAN-IP>/api/v1/policy/device-admin/policy-set
자격 증명	OpenAPI 계정 자격 증명을 사용합니다.
헤더	수락 : application/json Content-Type : application/json

2단계. 디바이스 관리 정책 집합 정보를 검색하는 데 사용되는 URL을 찾습니다.



API URI

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

ISE와 python 코드 예제를 실행 중인 디바이스 간의 양호한 연결을 보장합니다.

<#root>

```
from requests.auth import HTTPBasicAuth
import requests

requests.packages.urllib3.disable_warnings()

if __name__ == "__main__":

   url = "
https://10.106.33.92/api/v1/policy/device-admin/policy-set
"
    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())
```

이는 예상 출력의 예입니다.

Return Code: 200 Expected Outputs: {'version': '1.0.0', 'response': [{'default': True, 'id': '41ed8579-429b-42a8-879e-61861cb82bbf', 'name': 'Default', 'descr

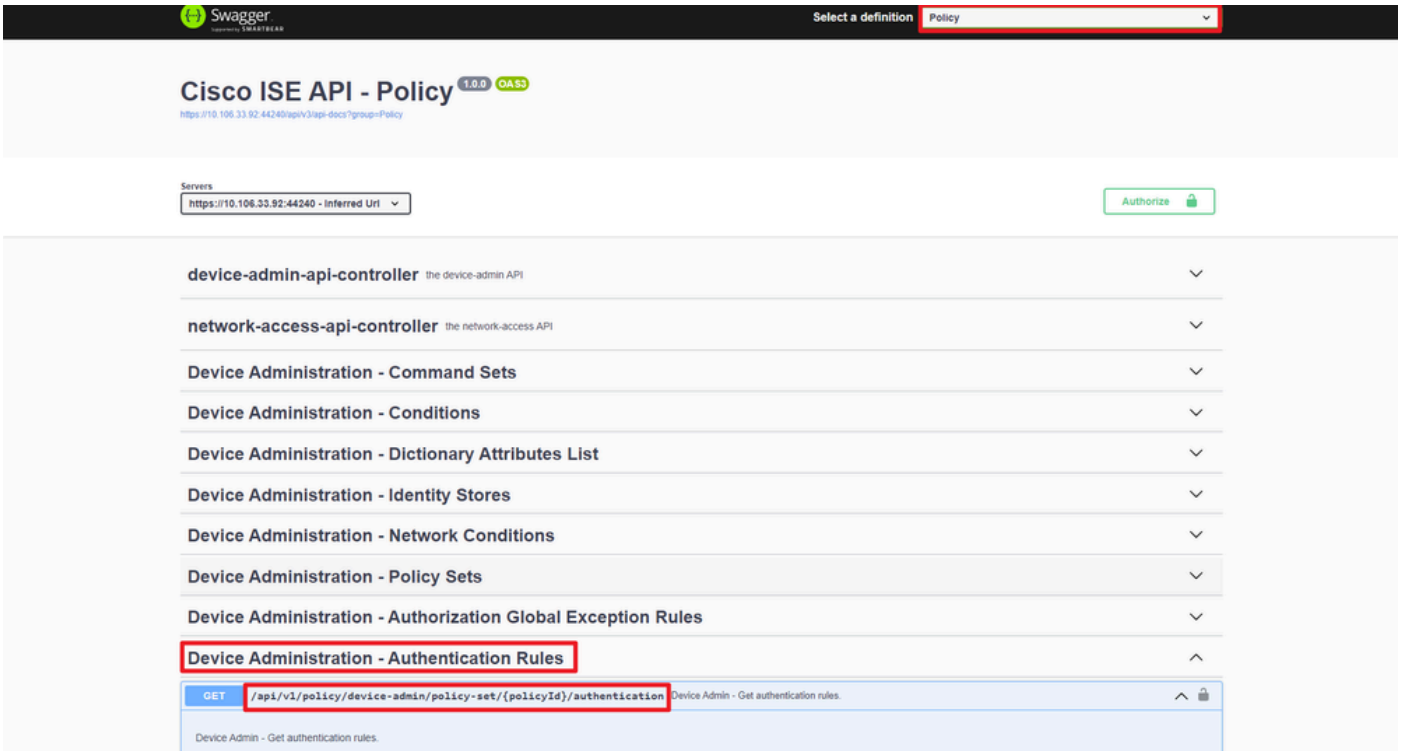
D장치 관리자 - 인증 규칙 가져오기

이 API는 특정 정책 집합의 인증 규칙을 검색합니다.

1단계. API 호출에 필요한 정보입니다.

방법	가져오기
URL	https://<ISE-PAN-IP>/api/v1/policy/device-admin/policy-set/<ID-Of-Policy-Set>/authentication
자격 증명	OpenAPI 계정 자격 증명을 사용합니다.
헤더	수락 : application/json Content-Type : application/json

2단계. 인증 규칙 정보를 검색하는 데 사용되는 URL을 찾습니다.



API URI

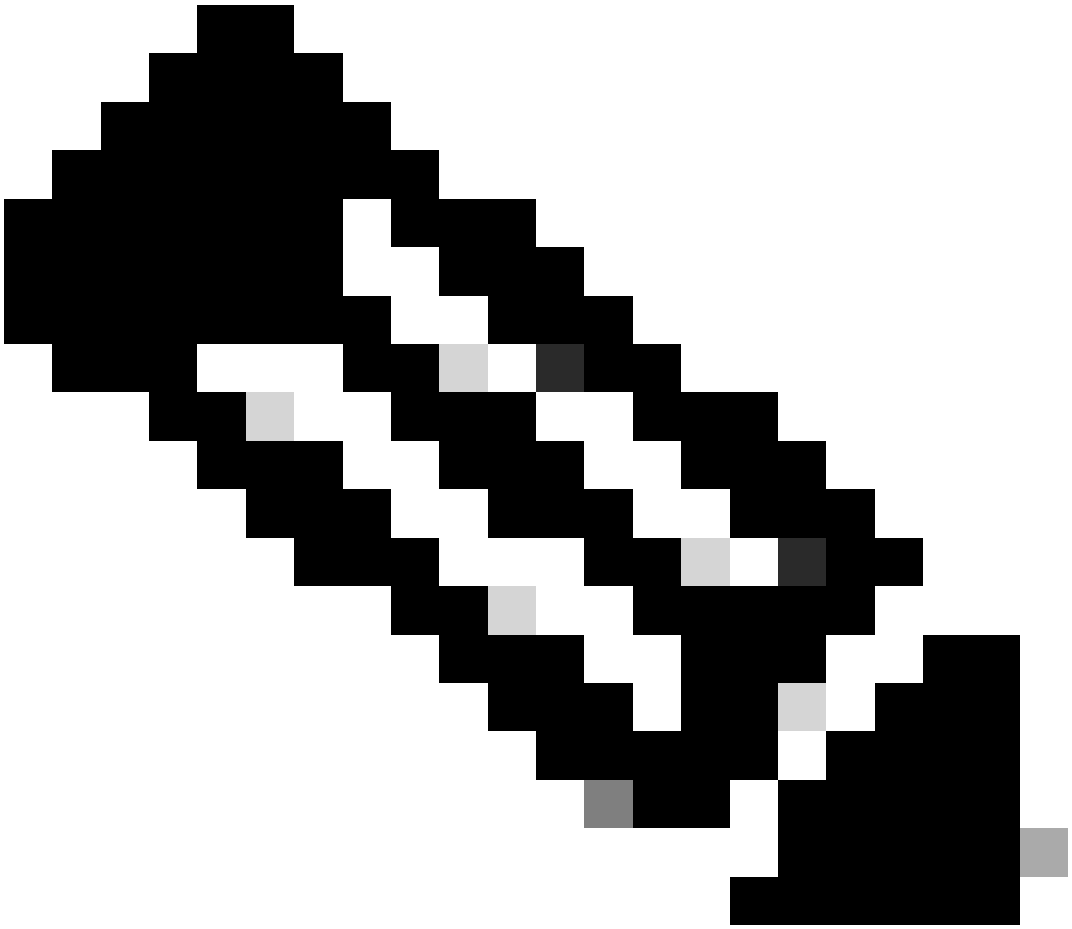
3단계. Python 코드의 예입니다. 내용을 복사하여 붙여넣습니다. ISE IP, 사용자 이름 및 비밀번호를 교체합니다. 실행할 python 파일로 저장합니다.

ISE와 python 코드 예제를 실행 중인 디바이스 간의 양호한 연결을 보장합니다.

<#root>

```
from requests.auth import HTTPBasicAuth
import requests

requests.packages.urllib3.disable_warnings()

if __name__ == "__main__":

   url = "
https://10.106.33.92/api/v1/policy/device-admin/policy-set/41ed8579-429b-42a8-879e-61861cb82bbf/authenti
"
    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())
```



참고: ID는 Device Admin(디바이스 관리) - List Of Policy Sets(정책 집합 목록)의 3단계에서 API 출력에서 가져옵니다. 예를 들어 41ed8579-429b-42a8-879e-61861cb82bbf는 TACACS 기본 정책 집합입니다.

이는 예상 출력의 예입니다.

Return Code: 200 Expected Outputs: {'version': '1.0.0', 'response': [{'rule': {'default': True, 'id': '73461597-0133-45ce-b4cb-6511ce56f262', 'name': 'Default'

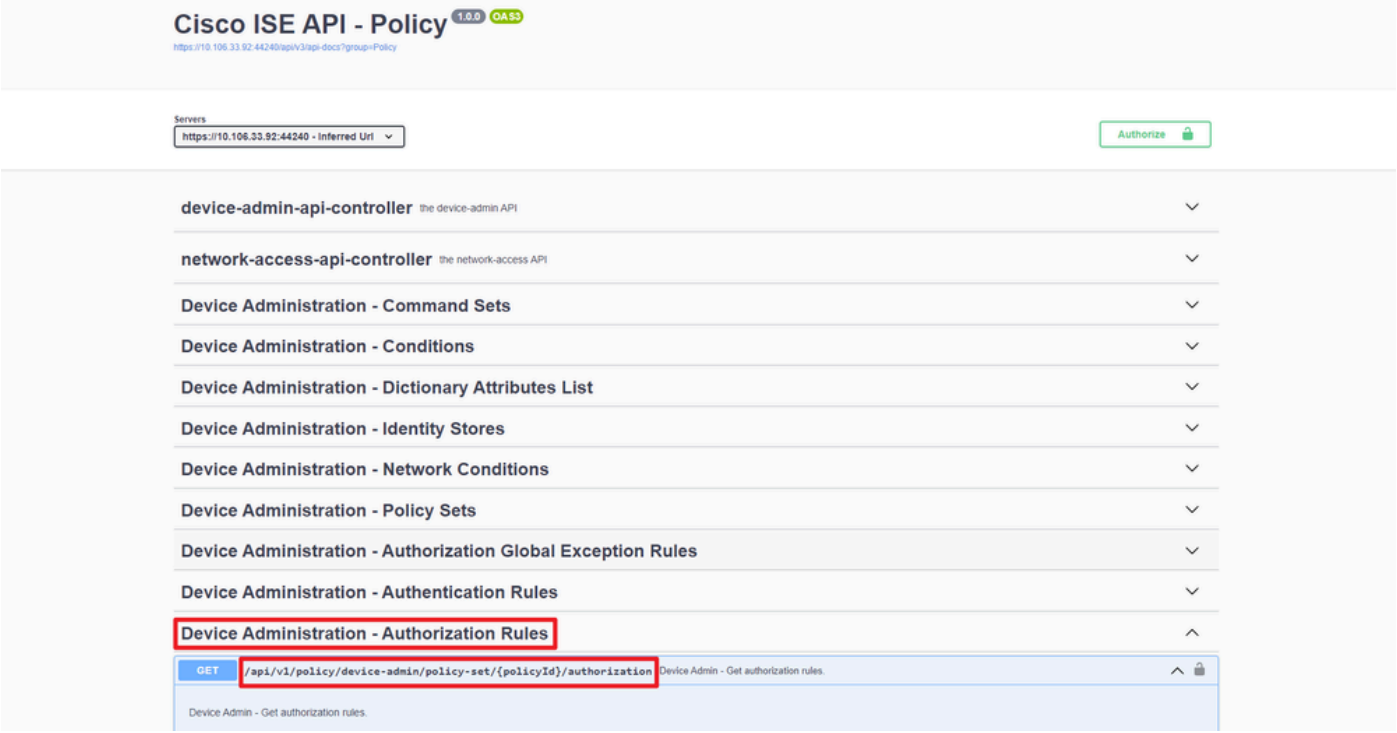
장치 관리자 - 권한 부여 규칙 가져오기

이 API는 특정 정책 세트의 권한 부여 규칙을 검색합니다.

1단계. API 호출에 필요한 정보입니다.

방법	가져오기
URL	https://<ISE-PAN-IP>/api/v1/policy/device-admin/policy-set/<ID-Of-Policy-Set>/authorization
자격 증명	OpenAPI 계정 자격 증명을 사용합니다.
헤더	수락 : application/json Content-Type : application/json

2단계. 권한 부여 규칙 정보를 검색하는 데 사용되는 URL을 찾습니다.



API URI

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

ISE와 python 코드 예제를 실행 중인 디바이스 간의 양호한 연결을 보장합니다.

<#root>

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

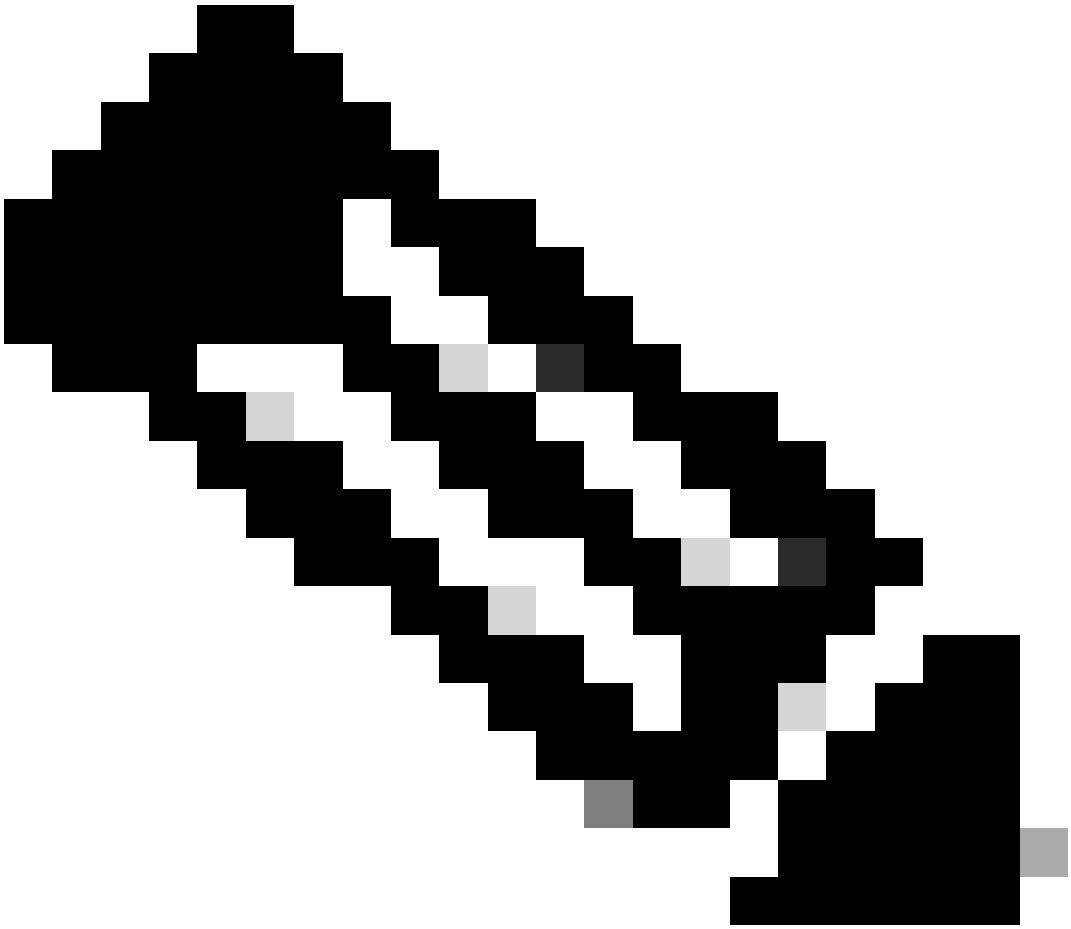
https://10.106.33.92/api/v1/policy/device-admin/policy-set/41ed8579-429b-42a8-879e-61861cb82bbf/authoriz

" 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 Device Admin(디바이스 관리) - List Of Policy Sets(정책 집합 목록)의 3단계에서 API 출력에서 가져옵니다. 예를 들어 41ed8579-429b-42a8-879e-61861cb82bbf는 TACACS 기본 정책 집합입니다.

이는 예상 출력의 예입니다.

Return Code:
200
Expected Outputs:
{'version': '1.0.0', 'response': [{'rule': {'default': True, 'id': '39d9f546-e58c-4f79-9856-c0a244b8a2ae', 'name': 'Default', 'hitCounts': 0, 'rank': 0, 'state': 'enable

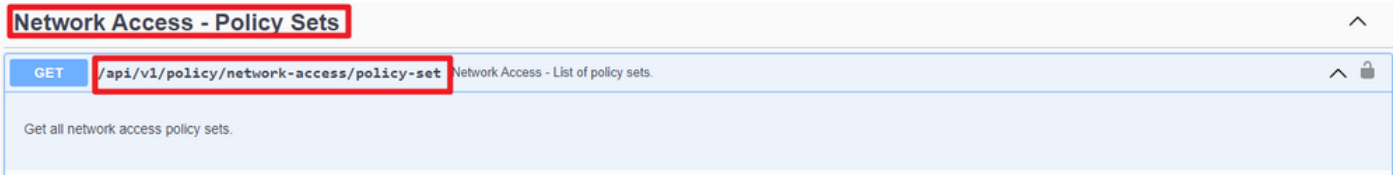
네트워크 액세스 - 정책 집합 목록

이 API는 ISE 구축의 네트워크 액세스 정책 집합을 검색합니다.

1단계. API 호출에 필요한 정보입니다.

방법	가져오기
URL	https://<ISE-PAN-IP>/api/v1/policy/network-access/policy-set
자격 증명	OpenAPI 계정 자격 증명을 사용합니다.
헤더	수락 : application/json Content-Type : application/json

2단계. 특정 ISE 노드 정보를 검색하는 데 사용되는 URL을 찾습니다.



API URI

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

ISE와 python 코드 예제를 실행 중인 디바이스 간의 양호한 연결을 보장합니다.

<#root>

```
from requests.auth import HTTPBasicAuth
import requests

requests.packages.urllib3.disable_warnings()

if __name__ == "__main__":

   url = "
https://10.106.33.92/api/v1/policy/network-access/policy-set
"
    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())
```

이는 예상 출력의 예입니다.

Return Code: 200 Expected Outputs: {'version': '1.0.0', 'response': [{'default': False, 'id': 'ba71a417-4a48-4411-8bc3-d5df9b115769', 'name': 'BGL_CFME0

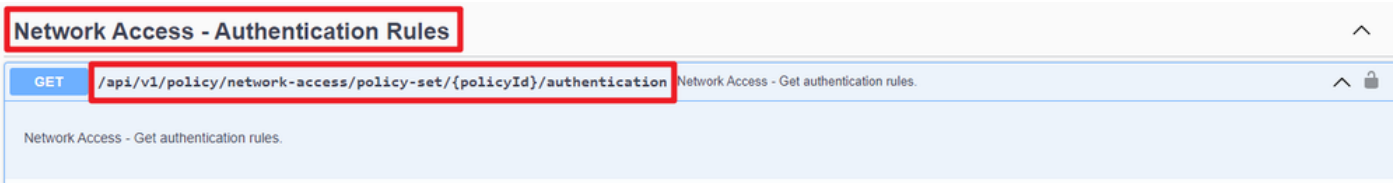
네트워크 액세스 - 인증 규칙 가져오기

이 API는 특정 정책 집합의 인증 규칙을 검색합니다.

1단계. API 호출에 필요한 정보입니다.

방법	가져오기
URL	https://<ISE-PAN-IP>/api/v1/policy/network-access/policy-set/<ID-Of-Policy-Set>/authentication
자격 증명	OpenAPI 계정 자격 증명을 사용합니다.
헤더	수락 : application/json Content-Type : application/json

2단계. 인증 규칙 정보를 검색하는 데 사용되는 URL을 찾습니다.



API URI

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

ISE와 python 코드 예제를 실행 중인 디바이스 간의 양호한 연결을 보장합니다.

<#root>

```
from requests.auth import HTTPBasicAuth
import requests

requests.packages.urllib3.disable_warnings()
```

if __name__ == "__main__":

url = "

https://10.106.33.92/api/v1/policy/network-access/policy-set/ba71a417-4a48-4411-8bc3-d5df9b115769/auther

"

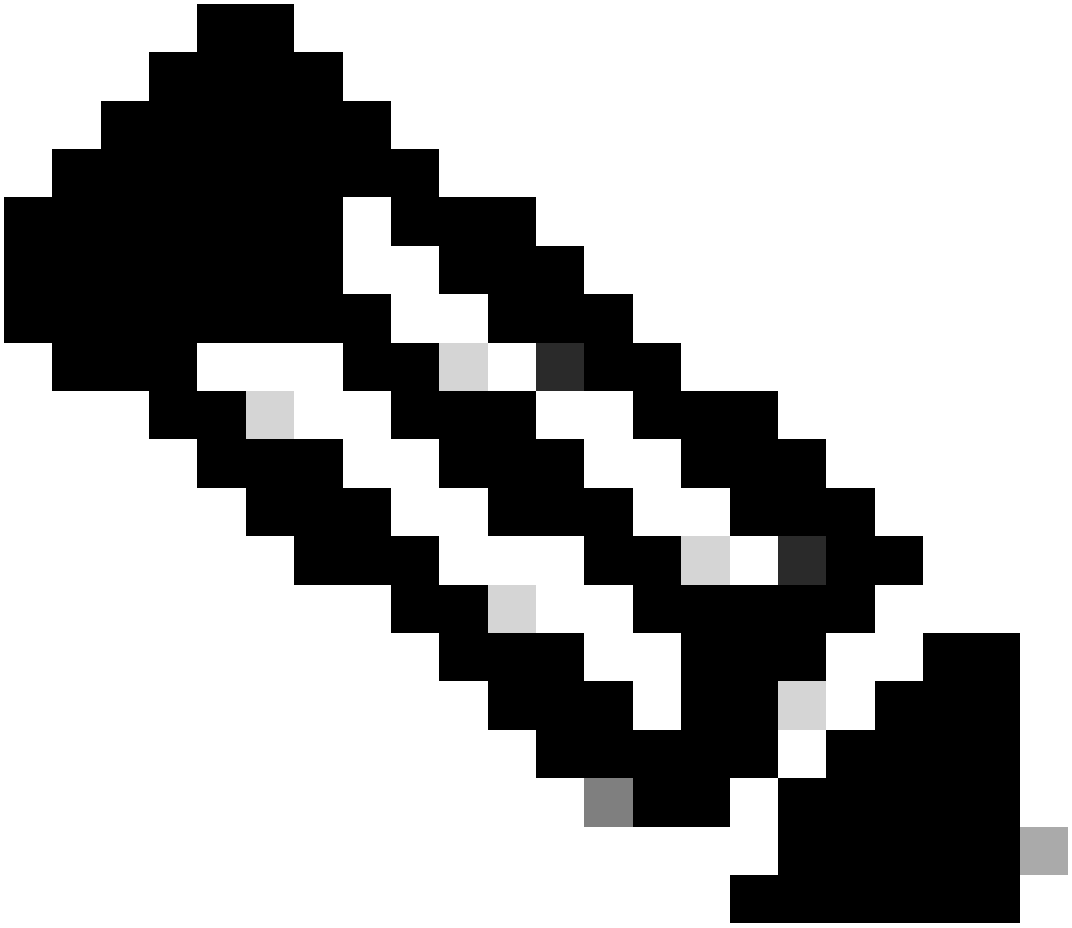
```
    headers = {
```

"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())
```



참고: ID는 Network Access - List Of Policy Sets(네트워크 액세스 - 정책 집합 목록) 3단계의 API 출력에서 가져옵니다. 예를 들어 ba71a417-4a48-4411-8bc3-d5df9b115769 는 입니다BGL_CFME02-FMC.

이는 예상 출력의 예입니다.

Return Code: 200 Expected Outputs: {'version': '1.0.0', 'response': [{'rule': {'default': True, 'id': '03875777-6c98-4114-a72e-a3e1651e533a', 'name': 'Default

네트워크 액세스 - 권한 부여 규칙 가져오기

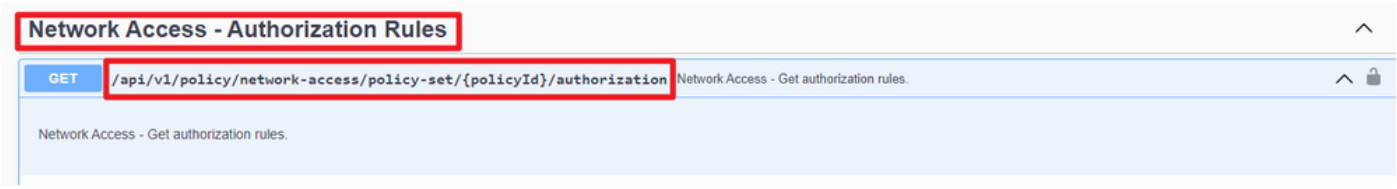
이 API는 특정 정책 세트의 권한 부여 규칙을 검색합니다.

1단계. API 호출에 필요한 정보입니다.

방법	가져오기

URL	https://<ISE-PAN-IP>/api/v1/policy/network-access/policy-set/<ID-Of-Policy-Set>/authorization
자격 증명	OpenAPI 계정 자격 증명을 사용합니다.
헤더	수락 : application/json Content-Type : application/json

2단계. 권한 부여 규칙 정보를 검색하는 데 사용되는 URL을 찾습니다.



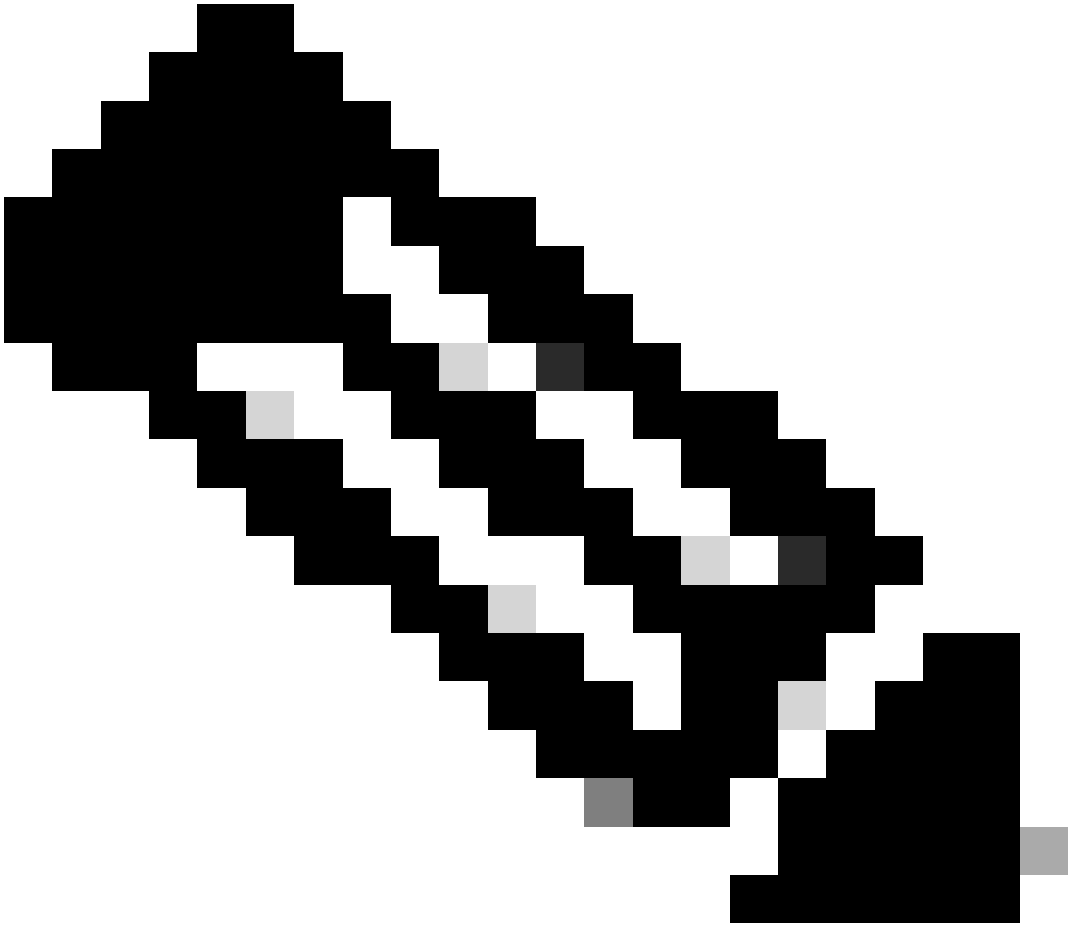
API URI

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

ISE와 python 코드 예제를 실행 중인 디바이스 간의 양호한 연결을 보장합니다.

<#root>

```
from requests.auth import HTTPBasicAuth
import requests

requests.packages.urllib3.disable_warnings()

if __name__ == "__main__":

   url = "
https://10.106.33.92/api/v1/policy/network-access/policy-set/ba71a417-4a48-4411-8bc3-d5df9b115769/author
"
    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())
```



참고: ID는 Network Access(네트워크 액세스) - List Of Policy Sets(정책 집합 목록)의 3단계에서 API 출력에서 가져옵니다. 예를 들어 ba71a417-4a48-4411-8bc3-d5df9b115769은 BGL_CFME02-FMC입니다.

이는 예상 출력의 예입니다.

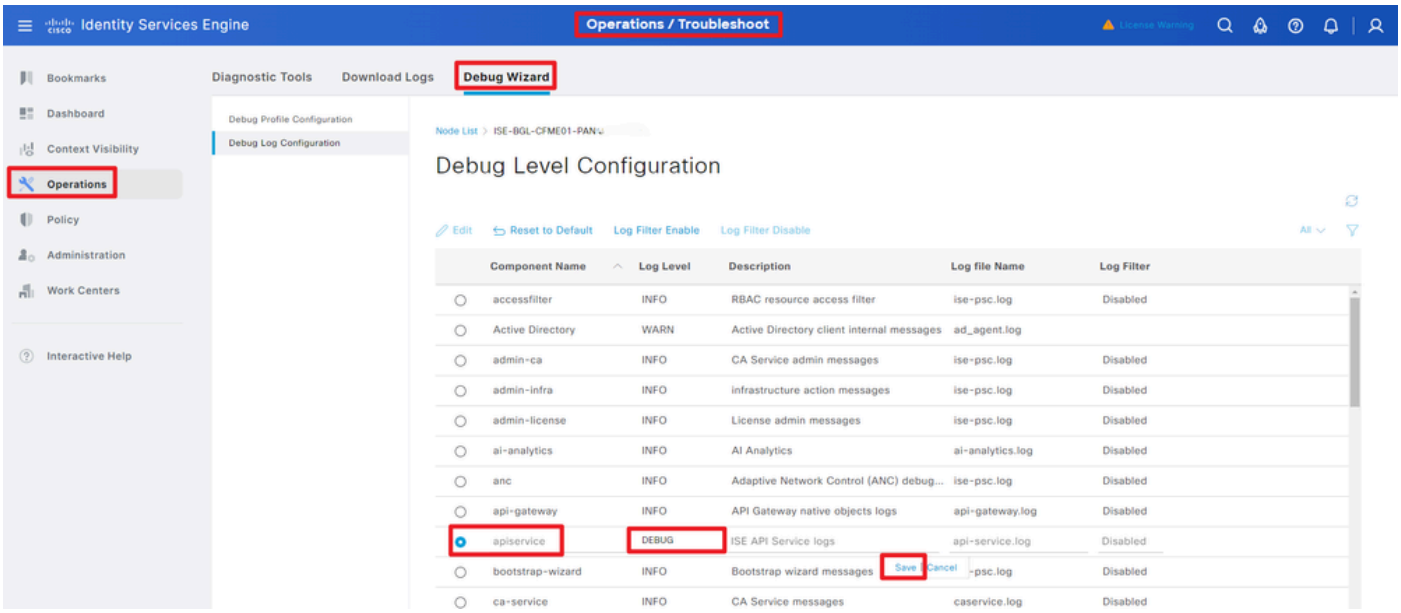
Return Code: 200 Expected Outputs: {'version': '1.0.0', 'response': [{'rule': {'default': False, 'id': 'bc67a4e5-9000-4645-9d75-7c2403ca22ac', 'name': 'FMC A

문제 해결

OpenAPI와 관련된 문제를 해결하려면 Debug Log Configuration(디버그 로그 컨피그레이션) 창에서 Theapiservicecomponent의 Log Level(로그 레벨)을 DEBUG로 설정합니다.

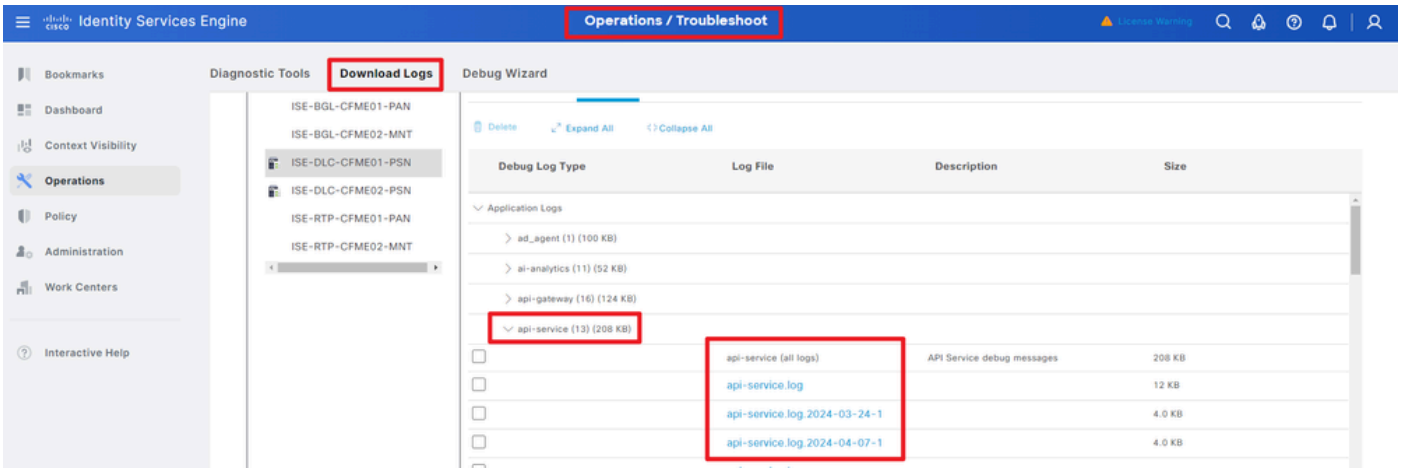
디버그를 활성화하려면 Operations(운영) > Troubleshoot(문제 해결) > Debug Wizard(디버그 마법

사) > Debug Log Configuration(디버그 로그 컨피그레이션) > ISE Node(ISE 노드) > apisservice(어플라이언스)로 이동합니다.



API 서비스 디버그

디버그 로그 파일을 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.