Use OpenAPI to Retrieve ISE Certificate Information on ISE 3.3

Contents

Introduction

This document describes the procedure for utilizing openAPI to manage Cisco Identity Services Engine (ISE) certificate.

Background

In the face of growing complexity in enterprise network security and management, Cisco ISE 3.1 introduces OpenAPI-formatted APIs that streamline certificate lifecycle management, offering a standardized and automated interface for efficient and secure certificate operations, helping administrators enforce strong security practices and maintain network compliance.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

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

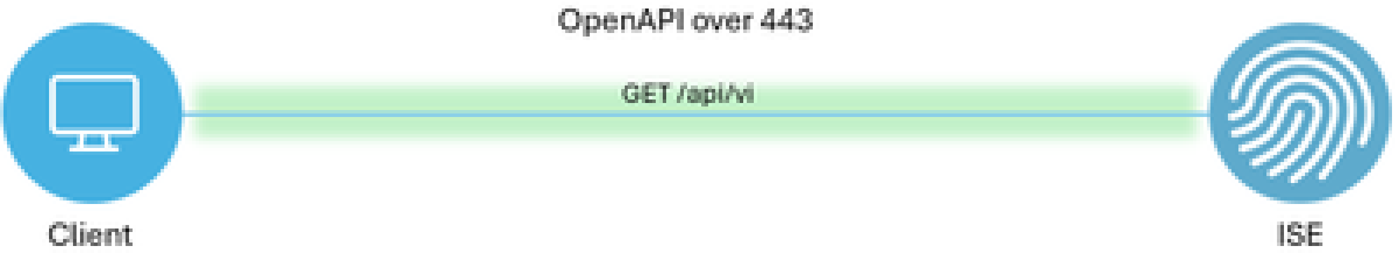
Components Used

- ISE 3.3
- Python 3.10.0

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Configure

Network Diagram

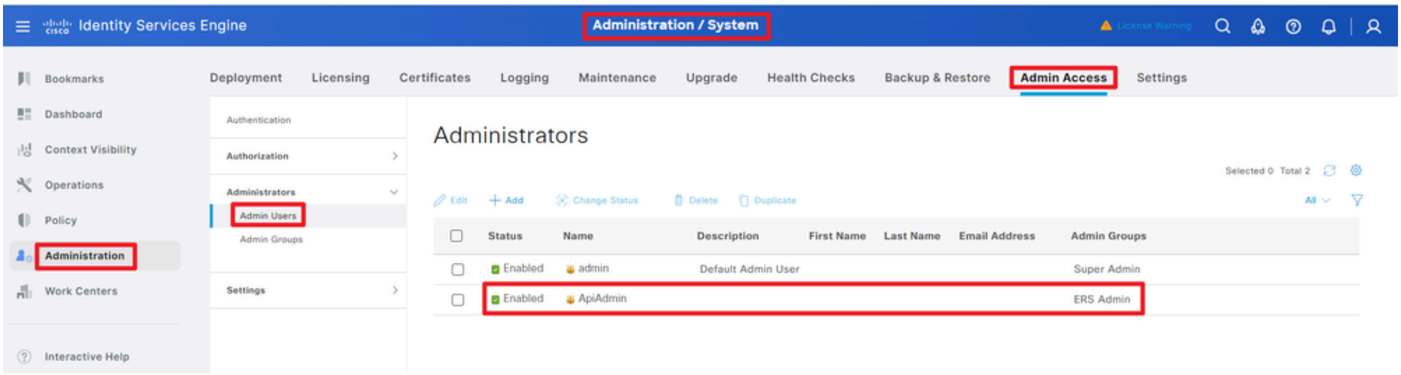


Topology

Configuration on ISE

Step 1: Add an Open API admin account

To add an API admin, navigate to**Administration** > **System** > **Admin Access** > **Administrators** > **Admin Users** > **Add.**



Administrators

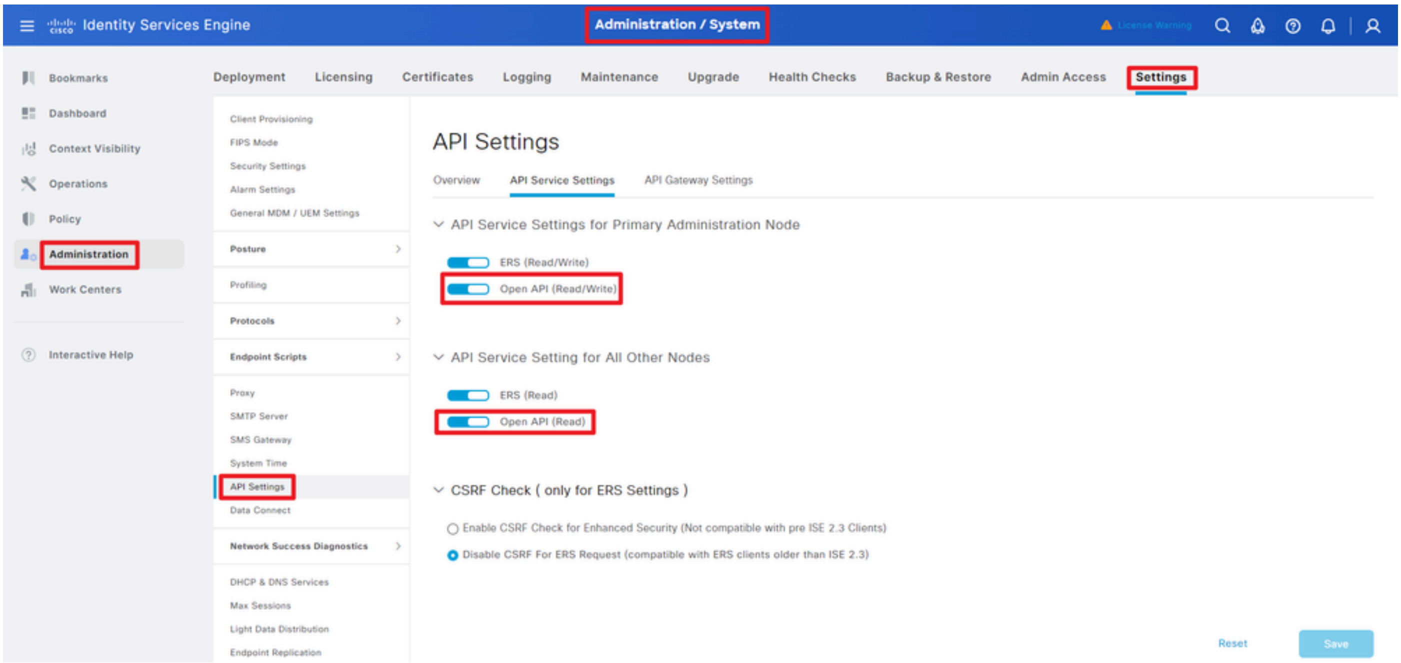
Selected 0 Total 2

□	Status	Name	Description	First Name	Last Name	Email Address	Admin Groups
□	Enabled	admin	Default Admin User				Super Admin
□	Enabled	ApiAdmin					ERS Admin

API Admin

Step 2: Enable Open API on ISE

Open API is disabled by default on ISE. To enable it, navigate to **Administration** > **System** > **Settings** > **API Settings** > **API Service Settings**. Toggle the Open API options. Click ***Save***.



API Settings

Overview | API Service Settings | API Gateway Settings

API Service Settings for Primary Administration Node

ERS (Read/Write)

Open API (Read/Write)

API Service Setting for All Other Nodes

ERS (Read)

Open API (Read)

CSRF Check (only for ERS Settings)

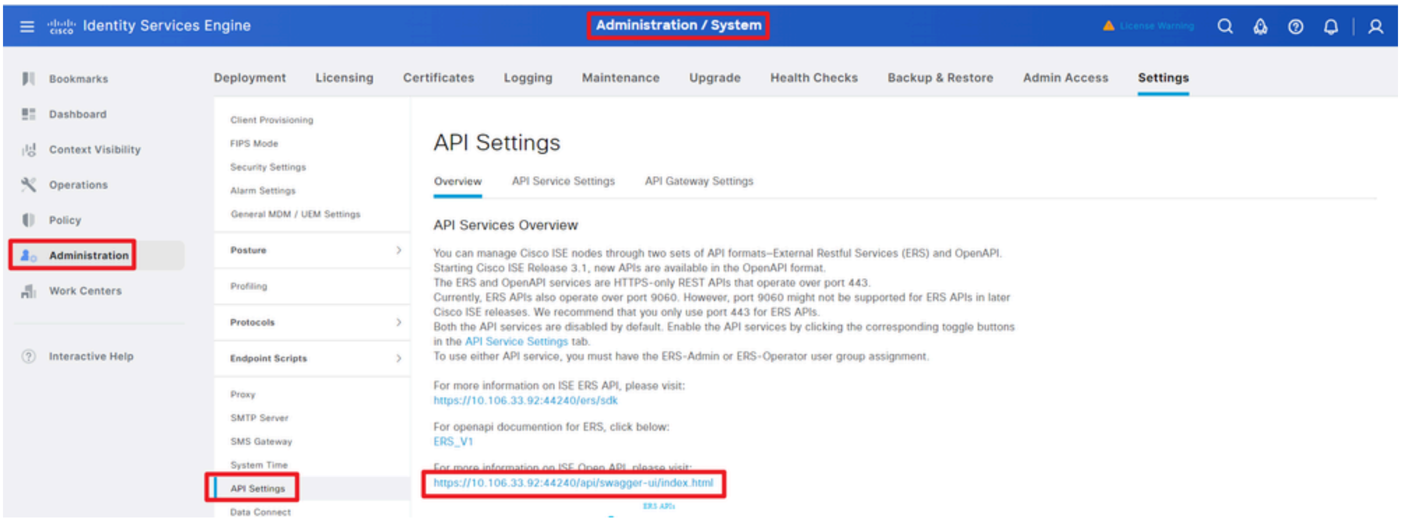
○ Enable CSRF Check for Enhanced Security (Not compatible with pre ISE 2.3 Clients)

● Disable CSRF For ERS Request (compatible with ERS clients older than ISE 2.3)

Enable OpenAPI

Step 3: Explore ISE open API

navigate to **Administration** > **System** > **Settings** > **API Settings** > **Overview**. Click open API visit link.



API Settings

Overview | API Service Settings | API Gateway Settings

API Services Overview

You can manage Cisco ISE nodes through two sets of API formats–External Restful Services (ERS) and OpenAPI.
Starting Cisco ISE Release 3.1, new APIs are available in the OpenAPI format.
The ERS and OpenAPI services are HTTPS-only REST APIs that operate over port 443.
Currently, ERS APIs also operate over port 9060. However, port 9060 might not be supported for ERS APIs in later Cisco ISE releases. We recommend that you only use port 443 for ERS APIs.
Both the API services are disabled by default. Enable the API services by clicking the corresponding toggle buttons in the API Service Settings tab.
To use either API service, you must have the ERS-Admin or ERS-Operator user group assignment.

For more information on ISE ERS API, please visit:
https://10.106.33.92:44240/ers/sdk

For openapi documention for ERS, click below:
ERS_V1

For more information on ISE Open API, please visit:
https://10.106.33.92:44240/api/swagger-ui/index.html

Visit OpenAPI

Python Examples

Get All System Certificates Of A Particular Node

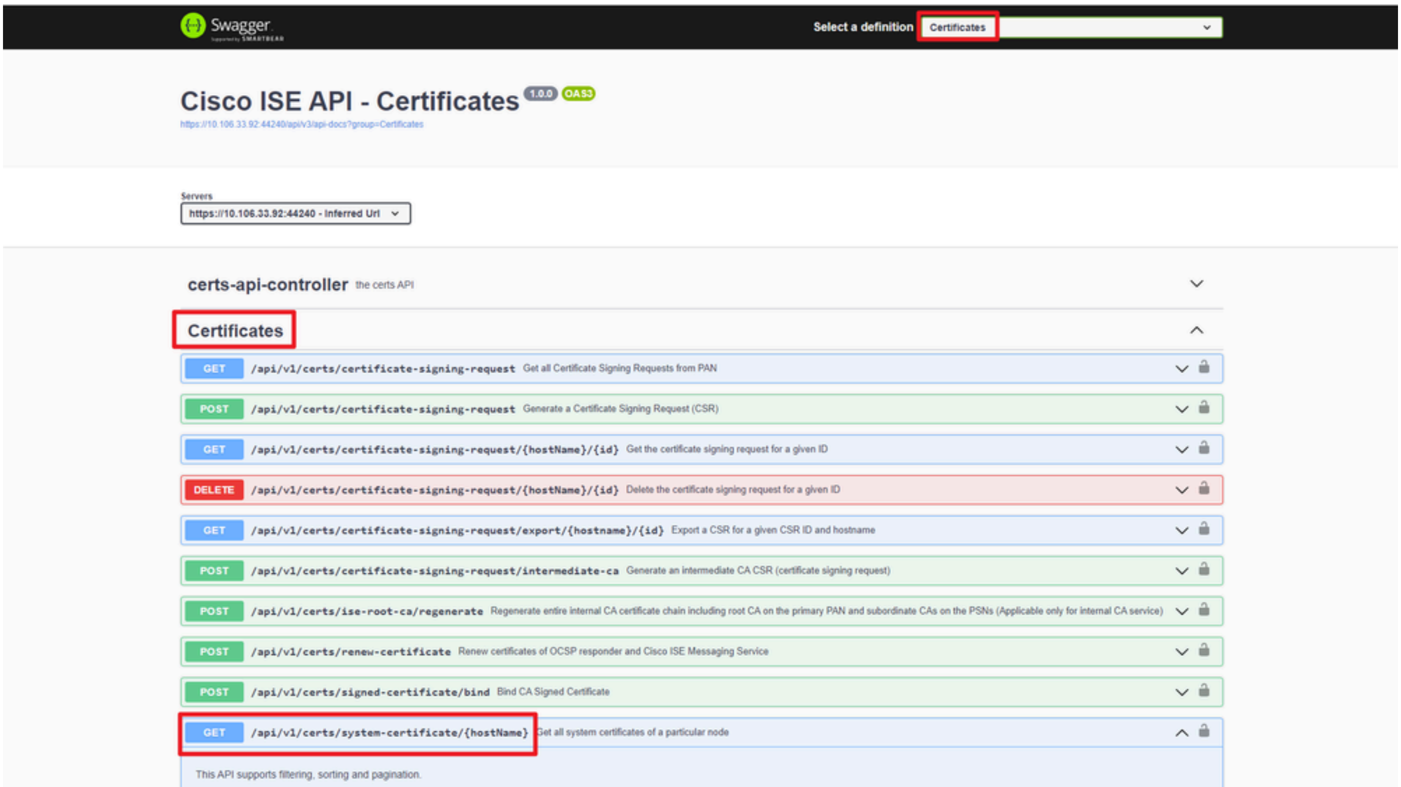
The API lists all the certificates of a particular ISE node.

Step 1: Required information for an API call.

Method	GET
URL	https://<ISE-PAN-IP>/api/v1/certs/system-certificate/<ISE-Node-Hostname>
Credentials	Use Open API account credentials

Headers	Accept : application/json Content-Type : application/json

Step 2: Locate the URL that is utilized to retrieve certificates of a particular ISE node.



Cisco ISE API - Certificates

https://10.106.33.92:44240/api/v3/api-docs?group=Certificates

certs-api-controller the certs API

Certificates

GET /api/v1/certs/certificate-signing-request Get all Certificate Signing Requests from PAN

POST /api/v1/certs/certificate-signing-request Generate a Certificate Signing Request (CSR)

GET /api/v1/certs/certificate-signing-request/{hostName}/{id} Get the certificate signing request for a given ID

DELETE /api/v1/certs/certificate-signing-request/{hostName}/{id} Delete the certificate signing request for a given ID

GET /api/v1/certs/certificate-signing-request/export/{hostname}/{id} Export a CSR for a given CSR ID and hostname

POST /api/v1/certs/certificate-signing-request/intermediate-ca Generate an intermediate CA CSR (certificate signing request)

POST /api/v1/certs/ise-root-ca/regenerate Regenerate entire internal CA certificate chain including root CA on the primary PAN and subordinate CAs on the PSNs (Applicable only for internal CA service)

POST /api/v1/certs/renew-certificate Renew certificates of OCSP responder and Cisco ISE Messaging Service

POST /api/v1/certs/signed-certificate/bind Bind CA Signed Certificate

GET /api/v1/certs/system-certificate/{hostName} Get all system certificates of a particular node

This API supports filtering, sorting and pagination.

API URI

Step 3: Here is the example of Python Code. Copy and paste the content. Replace the ISE IP, username, password. Save as a python file to execute.

Ensure the good connectivity between ISE and the device running the python code example.

<#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())
```

Here is the example of expected outputs.

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

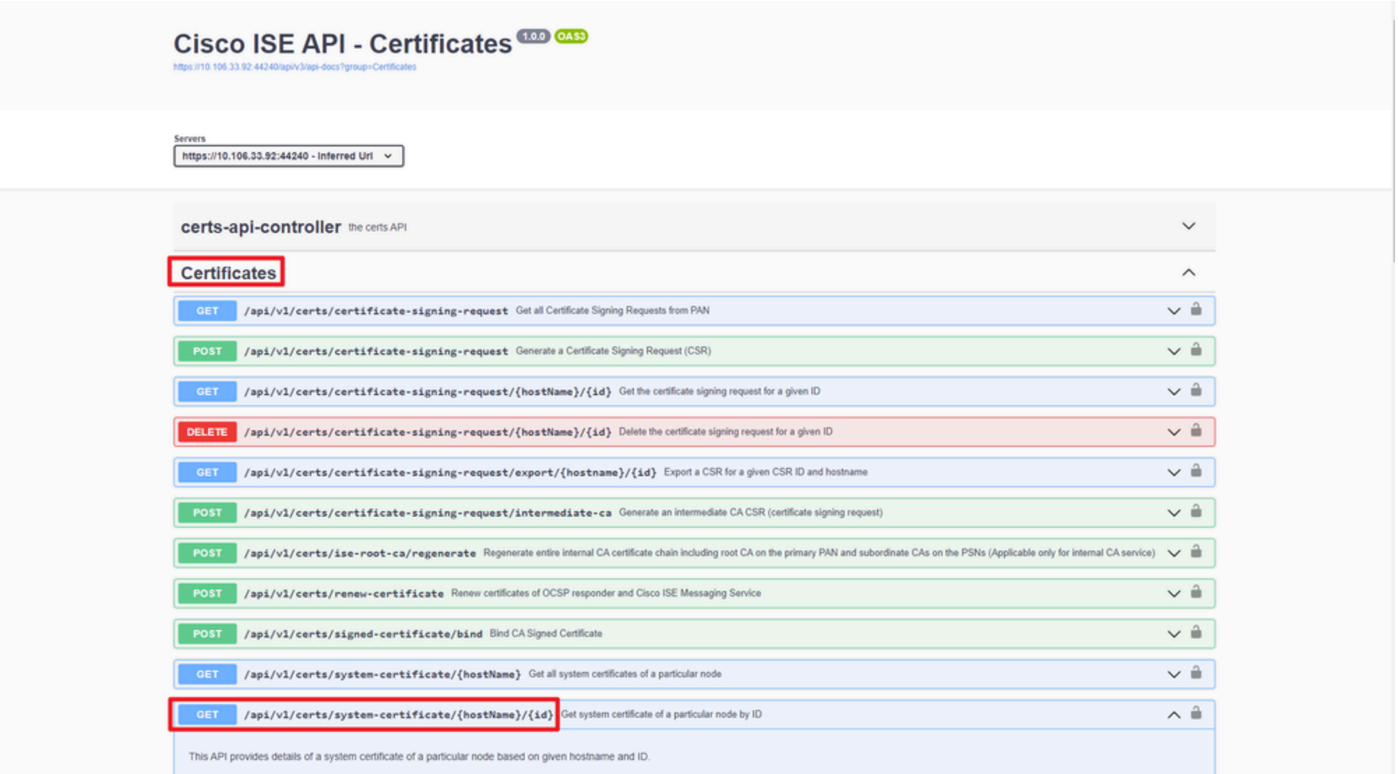
Get System Certificate Of A Particular Node By ID

This API provides details of a system certificate of a particular node based on given hostname and ID.

Step 1: Required information for an API call.

Method	GET
URL	https://<ISE-PAN-IP>/api/v1/certs/system-certificate/<ISE-Node-Hostname>/<ID-Of-Certificate>
Credentials	Use Open API account credentials
Headers	Accept : application/json Content-Type : application/json

Step 2: Locate the URL that is utilized to retrieve the certificate of a particular node based on given hostname and ID.



Cisco ISE API - Certificates

https://10.106.33.92:44240/api/v3/api-docs?group=Certificates

Servers: https://10.106.33.92:44240 - Inferred Url

certs-api-controller the certs API

Certificates

```
GET     /api/v1/certs/certificate-signing-request  Get all Certificate Signing Requests from PAN
POST    /api/v1/certs/certificate-signing-request  Generate a Certificate Signing Request (CSR)
GET     /api/v1/certs/certificate-signing-request/{hostName}/{id}  Get the certificate signing request for a given ID
DELETE  /api/v1/certs/certificate-signing-request/{hostName}/{id}  Delete the certificate signing request for a given ID
GET     /api/v1/certs/certificate-signing-request/export/{hostname}/{id}  Export a CSR for a given CSR ID and hostname
POST    /api/v1/certs/certificate-signing-request/intermediate-ca  Generate an intermediate CA CSR (certificate signing request)
POST    /api/v1/certs/ise-root-ca/regenerate  Regenerate entire internal CA certificate chain including root CA on the primary PAN and subordinate CAs on the PSNs (Applicable only for internal CA service)
POST    /api/v1/certs/renew-certificate  Renew certificates of OCSP responder and Cisco ISE Messaging Service
POST    /api/v1/certs/signed-certificate/bind  Bind CA Signed Certificate
GET     /api/v1/certs/system-certificate/{hostName}  Get all system certificates of a particular node
GET     /api/v1/certs/system-certificate/{hostName}/{id}  Get system certificate of a particular node by ID
```

This API provides details of a system certificate of a particular node based on given hostname and ID.

API URI

Step 3: Here is the example of Python Code. Copy and paste the content. Replace the ISE IP, username, password. Save as a python file to execute.

Ensure the good connectivity between ISE and the device running the python code example.

<#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:")
    print(response.status_code)
    print("Expected Outputs:")
    print(response.json())
```

Note: The ID is from API outputs in step 3 of "Get All System Certificates Of A Particular Node", for example, 5b5b28e4-2a51-495c-8413-610190e1070b is "Default self-signed saml server certificate - CN=SAML_ISE-DLC-CFME02-PSN.cisco.com".

Here is the example of expected outputs.

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

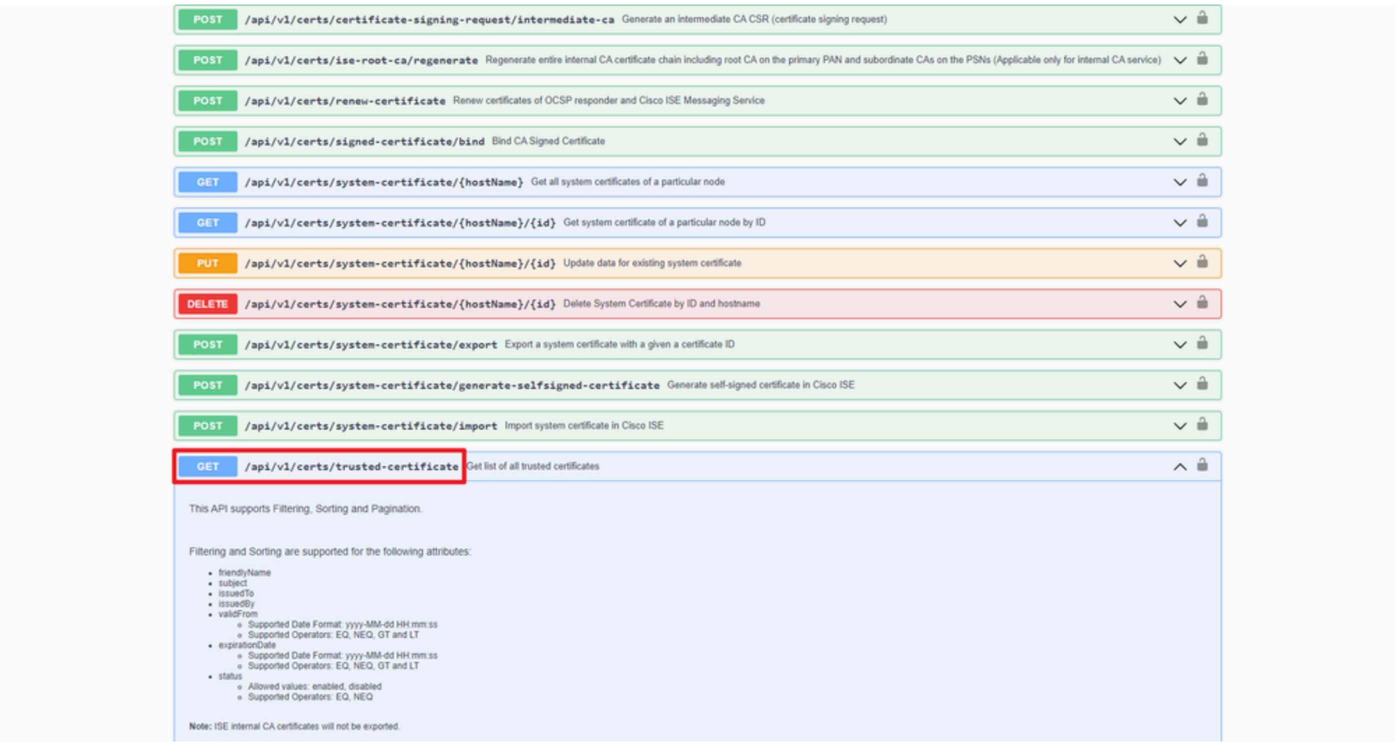
Get List Of All Trusted Certificates

The API lists all the trusted certificates of ISE cluster.

Step 1: Required information for an API call.

Method	GET
URL	https://<ISE-PAN-IP>/api/v1/certs/trusted-certificate
Credentials	Use Open API account credentials
Headers	Accept : application/json Content-Type : application/json

Step 2: Locate the URL that is utilized to retrieve trusted certificates.



POST `/api/v1/certs/certificate-signing-request/intermediate-ca` Generate an intermediate CA CSR (certificate signing request)

POST `/api/v1/certs/ise-root-ca/regenerate` Regenerate entire internal CA certificate chain including root CA on the primary PAN and subordinate CAs on the PSNs (Applicable only for internal CA service)

POST `/api/v1/certs/renew-certificate` Renew certificates of OCSP responder and Cisco ISE Messaging Service

POST `/api/v1/certs/signed-certificate/bind` Bind CA Signed Certificate

GET `/api/v1/certs/system-certificate/{hostName}` Get all system certificates of a particular node

GET `/api/v1/certs/system-certificate/{hostName}/{id}` Get system certificate of a particular node by ID

PUT `/api/v1/certs/system-certificate/{hostName}/{id}` Update data for existing system certificate

DELETE `/api/v1/certs/system-certificate/{hostName}/{id}` Delete System Certificate by ID and hostname

POST `/api/v1/certs/system-certificate/export` Export a system certificate with a given a certificate ID

POST `/api/v1/certs/system-certificate/generate-selfsigned-certificate` Generate self-signed certificate in Cisco ISE

POST `/api/v1/certs/system-certificate/import` Import system certificate in Cisco ISE

GET `/api/v1/certs/trusted-certificate` Get list of all trusted certificates

This API supports Filtering, Sorting and Pagination.

Filtering and Sorting are supported for the following attributes:

- friendlyName
- subject
- issuedTo
- issuedBy
- validFrom
  - Supported Date Format: yyyy-MM-dd HH:mm:ss
  - Supported Operators: EQ, NEQ, GT and LT
- expirationDate
  - Supported Date Format: yyyy-MM-dd HH:mm:ss
  - Supported Operators: EQ, NEQ, GT and LT
- status
  - Allowed values: enabled, disabled
  - Supported Operators: EQ, NEQ

Note: ISE internal CA certificates will not be exported.

API URI

Step 3: Here is the example of Python Code. Copy and paste the content. Replace the ISE IP, username, password. Save as a python file to execute.

Ensure the good connectivity between ISE and the device running the python code example.

<#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:")
    print(response.status_code)
    print("Expected Outputs:")
    print(response.json())
```

Here is the example of expected outputs.(Omitted)

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

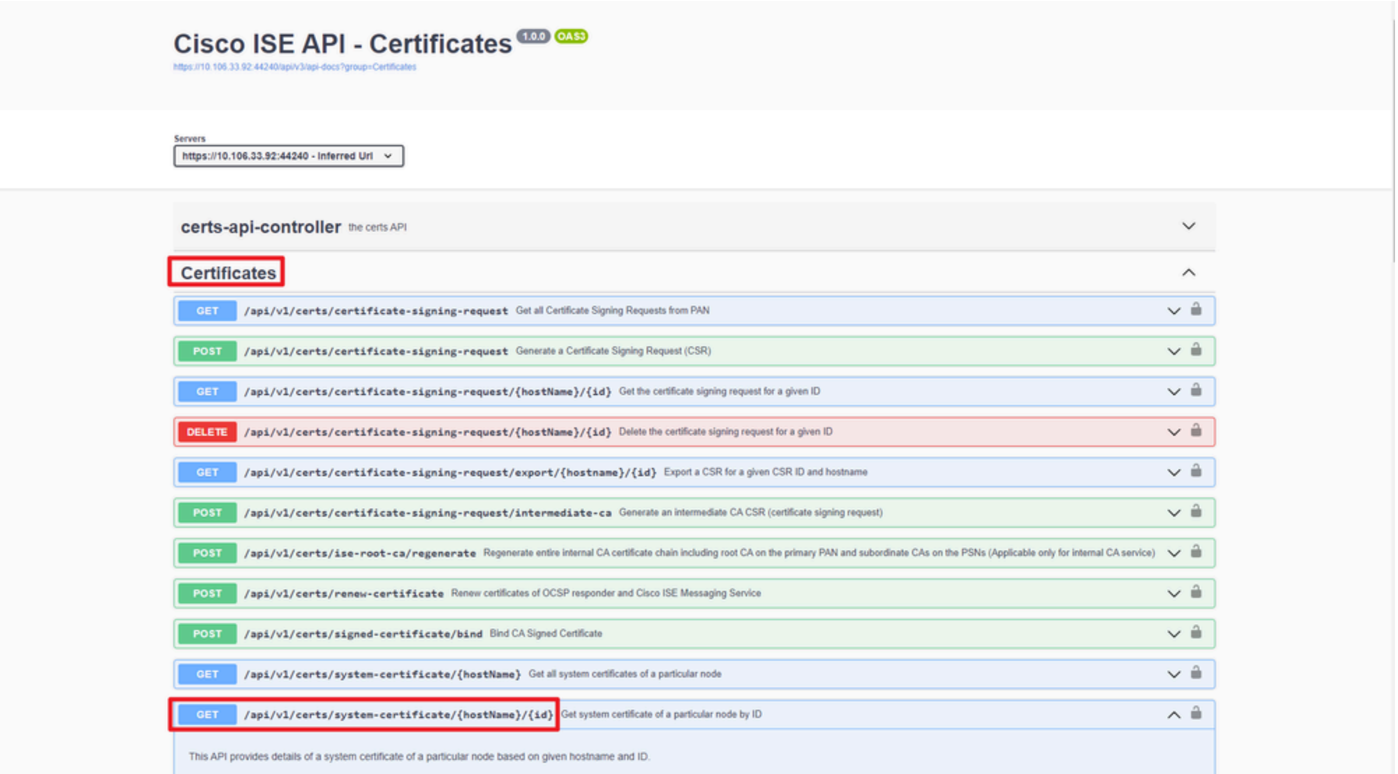
Get Trust Certificate By ID

This API can displays details of a Trust Certificate based on a given ID.

Step 1: Required information for an API call.

Method	GET
URL	https://<ISE-PAN-IP>/api/v1/certs/trusted-certificate/<ID-Of-Certificate>
Credentials	Use Open API account credentials
Headers	Accept : application/json Content-Type : application/json

Step 2: Locate the URL that is utilized to retrieve deployment information.



Cisco ISE API - Certificates 1.0.0 OAS3

https://10.106.33.92:44240/api/v3/api-docs?group=Certificates

Servers

https://10.106.33.92:44240 - Inferred Url

certs-api-controller the certs API

Certificates

GET /api/v1/certs/certificate-signing-request Get all Certificate Signing Requests from PAN

POST /api/v1/certs/certificate-signing-request Generate a Certificate Signing Request (CSR)

GET /api/v1/certs/certificate-signing-request/{hostName}/{id} Get the certificate signing request for a given ID

DELETE /api/v1/certs/certificate-signing-request/{hostName}/{id} Delete the certificate signing request for a given ID

GET /api/v1/certs/certificate-signing-request/export/{hostname}/{id} Export a CSR for a given CSR ID and hostname

POST /api/v1/certs/certificate-signing-request/intermediate-ca Generate an intermediate CA CSR (certificate signing request)

POST /api/v1/certs/ise-root-ca/regenerate Regenerate entire internal CA certificate chain including root CA on the primary PAN and subordinate CAs on the PSNs (Applicable only for internal CA service)

POST /api/v1/certs/renew-certificate Renew certificates of OCSP responder and Cisco ISE Messaging Service

POST /api/v1/certs/signed-certificate/bind Bind CA Signed Certificate

GET /api/v1/certs/system-certificate/{hostName} Get all system certificates of a particular node

GET /api/v1/certs/system-certificate/{hostName}/{id} Get system certificate of a particular node by ID

This API provides details of a system certificate of a particular node based on given hostname and ID.

API URI

Step 3: Here is the example of Python Code. Copy and paste the content. Replace the ISE IP, username, password. Save as a python file to execute.

Ensure the good connectivity between ISE and the device running the python code example.

<#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:")
    print(response.status_code)
    print("Expected Outputs:")
    print(response.json())
```

Note: The ID is from API outputs in step 3 of "Get List Of All Trusted Certificates", for example, 147d97cc-6ce9-43d7-9928-8cd0fa83e140 is "VeriSign Class 3 Public Primary Certification Authority".

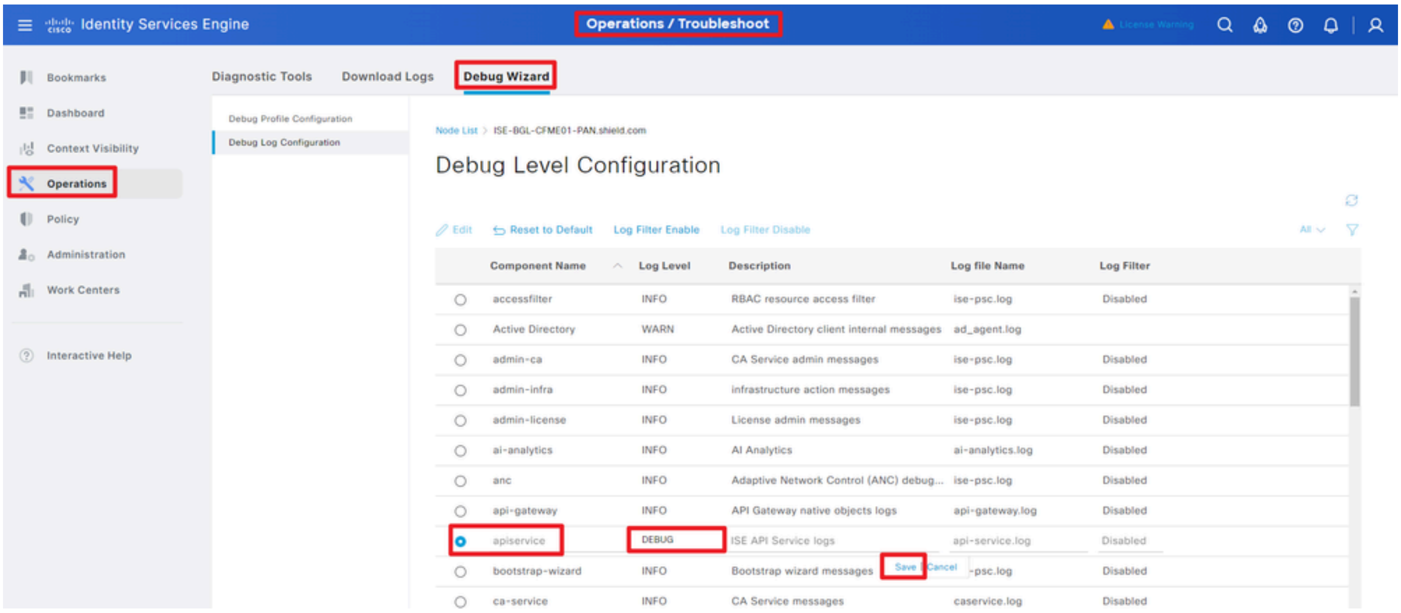
Here is the example of expected outputs.

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

Troubleshoot

To troubleshoot issues that are related to the Open APIs, set the**Log Level**for the**apiservice**component to**DEBUG**in the**Debug Log Configuration**window.

To enable debug, Navigate to **Operations > Troubleshoot > Debug Wizard > Debug Log Configuration > ISE Node > apiservice.**

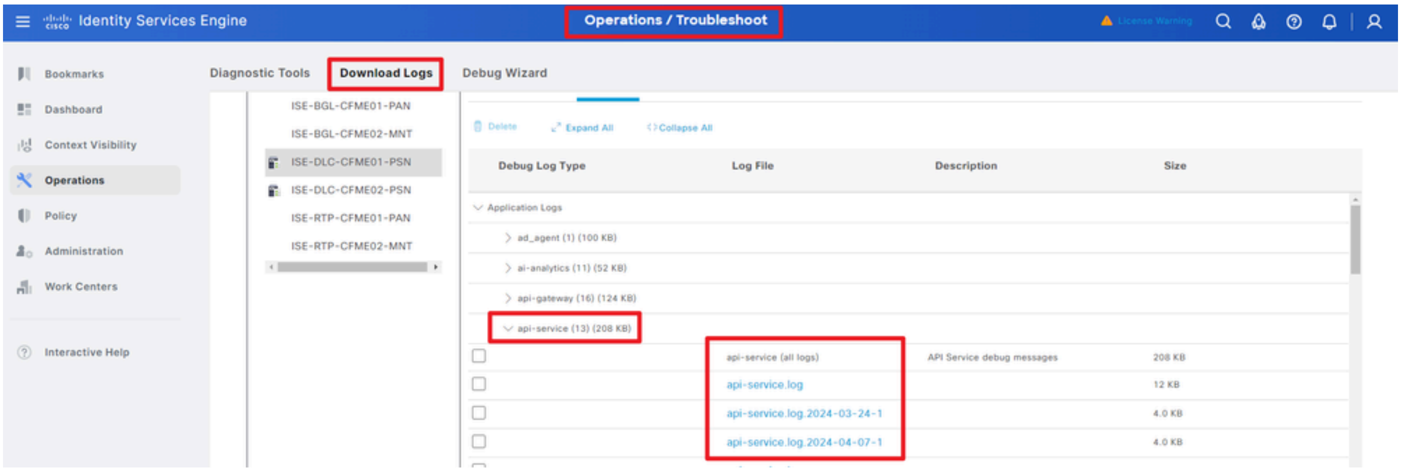


Debug Level Configuration

	Component Name	Log Level	Description	Log file Name	Log Filter
○	accessfilter	INFO	RBAC resource access filter	ise-psc.log	Disabled
○	Active Directory	WARN	Active Directory client internal messages	ad_agent.log	
○	admin-ca	INFO	CA Service admin messages	ise-psc.log	Disabled
○	admin-infra	INFO	infrastructure action messages	ise-psc.log	Disabled
○	admin-license	INFO	License admin messages	ise-psc.log	Disabled
○	ai-analytics	INFO	AI Analytics	ai-analytics.log	Disabled
○	anc	INFO	Adaptive Network Control (ANC) debug...	ise-psc.log	Disabled
○	api-gateway	INFO	API Gateway native objects logs	api-gateway.log	Disabled
●	apiservice	DEBUG	ISE API Service logs	api-service.log	Disabled
○	bootstrap-wizard	INFO	Bootstrap wizard messages	-psc.log	Disabled
○	ca-service	INFO	CA Service messages	caservice.log	Disabled

API Service Debug

To download debug logs, Navigate to **Operations > Troubleshoot > Downlaod Logs > ISE PAN Node > Debug Logs.**



	Debug Log Type	Log File	Description	Size
	Application Logs			
	ad_agent (1) (100 KB)			
	ai-analytics (11) (52 KB)			
	api-gateway (16) (124 KB)			
	api-service (13) (208 KB)			
□		api-service (all logs)	API Service debug messages	208 KB
□		api-service.log		12 KB
□		api-service.log.2024-03-24-1		4.0 KB
□		api-service.log.2024-04-07-1		4.0 KB

Download Debug Logs