



สำนักคอมพิวเตอร์
Computer Center



**Intermediate Certificate Authority API Server
For Digital Signature Usage**

Manual

Date

5 July 2023

Version

0.1

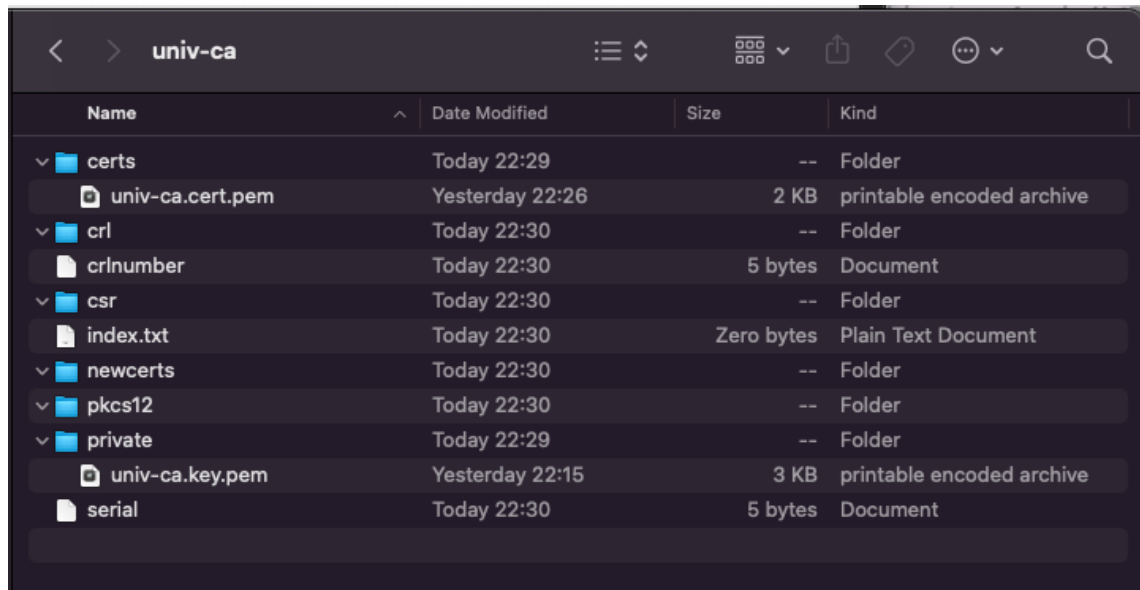
Innovative Co., Ltd
contact@innovative.co.th

Requirement

- **Public IP** *หากต้องการใช้งาน Certificate Authority นอก LAN / ผ่านอินเทอร์เน็ต
- **Docker**
- **Folder univ-ca UU Docker Host**

ซึ่งสร้างตามคู่มือของทาง Thai University Consortium ถึงข้อ 3. และทาง Thai University Consortium ได้ออก Certificate สำหรับให้ทางมหาวิทยาลัยเป็น Intermediate Certificate Authority ให้ใช้งานเรียบร้อยแล้ว

ตัวอย่าง Folder univ-ca ที่ถูกต้องพร้อมใช้งาน



Name	Date Modified	Size	Kind
certs	Today 22:29	--	Folder
univ-ca.cert.pem	Yesterday 22:26	2 KB	printable encoded archive
crl	Today 22:30	--	Folder
crlnumber	Today 22:30	5 bytes	Document
csr	Today 22:30	--	Folder
index.txt	Today 22:30	Zero bytes	Plain Text Document
newcerts	Today 22:30	--	Folder
pkcs12	Today 22:30	--	Folder
private	Today 22:29	--	Folder
univ-ca.key.pem	Yesterday 22:15	3 KB	printable encoded archive
serial	Today 22:30	5 bytes	Document

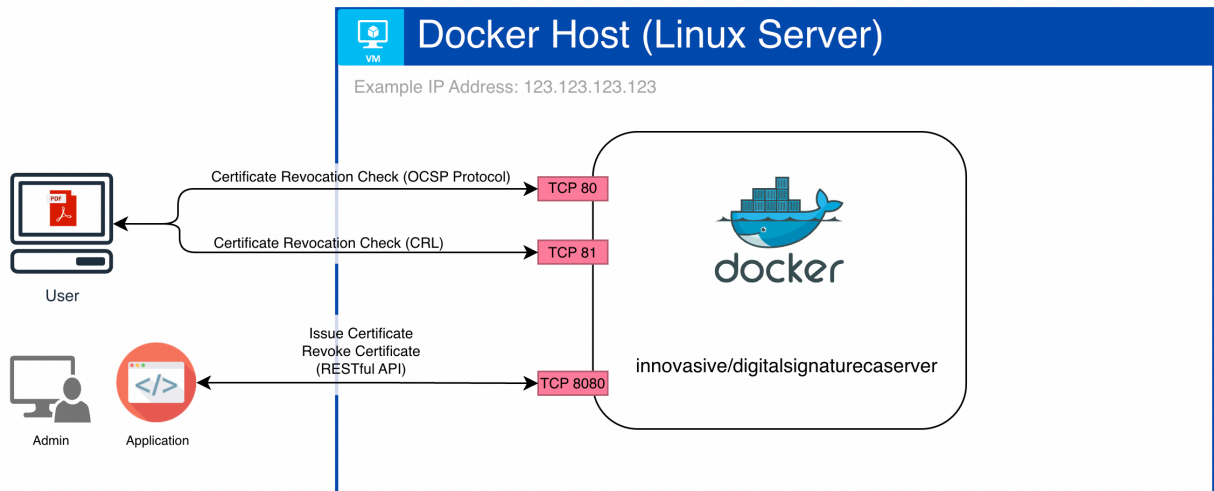
ไฟล์ที่จำเป็นได้แก่

- /univ-ca/certs/univ-ca.cert.pem คือไฟล์ Certificate ที่ทาง Thai University Consortium ได้ทำการ Signed ส่งกลับมาให้จาก CSR ที่ทำการส่งไป
 - /univ-ca/private/univ-ca.key.pem คือไฟล์ Private Key ที่เราสร้างตามคู่มือ
- โดยระบบจะเช็คว่ามีสองไฟล์นี้ และสองไฟล์นี้ Match กันจึงจะ Start Container เพื่อใช้งานได้สำเร็จ ส่วนไฟล์และ Folder อื่นๆ หากไม่มีไม่เป็นไร ระบบจะสร้างให้เองตอน Start Container

Docker Image

<https://hub.docker.com/r/innovasive/digitalsignaturecaserver>

Services Port on Docker Image



TCP Port 80

สำหรับให้บริการ Service สำหรับ User / Client เช็คว่า Certificate ที่ออกไปจาก Certificate Authority นี้ถูกเพิกถอน (Revoke) แล้วหรือยังด้วย Online Certificate Status Protocol (OCSP)

TCP Port 81

สำหรับให้บริการ Service สำหรับ User / Client เช็คว่า Certificate ที่ออกไปจาก Certificate Authority นี้ถูกเพิกถอน (Revoke) แล้วหรือยังด้วย Certificate Revocation List (CRL)

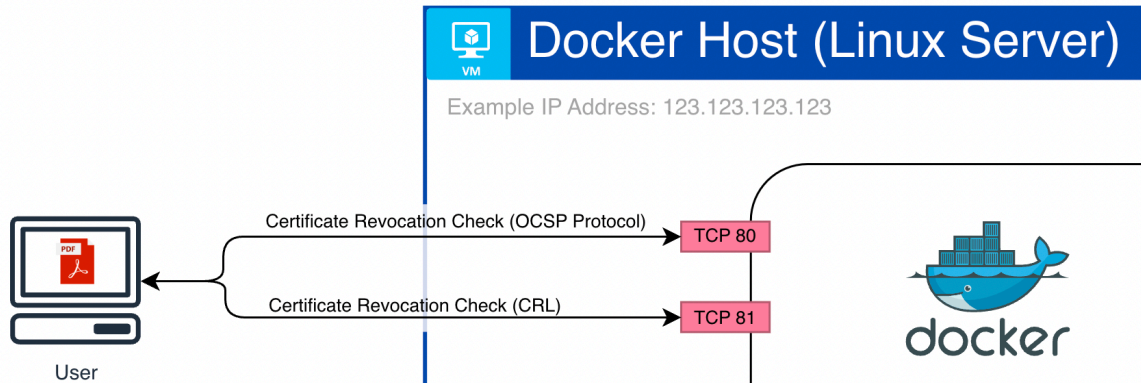
TCP Port 8080

สำหรับให้ Application หรือ Admin เชื่อมต่อ ผ่านทาง API โดยจะสามารถควบคุม Certificate Authority นี้ได้ผ่านทาง Restful API โดยอ้างอิงการใช้งานได้จากเอกสาร API Documentation

Config Docker Image (CA) By .env File

- การตั้งค่า CRL และ OCSP URL

การตั้งค่าจำเป็นจะต้องตั้งค่าก่อนการใช้งาน Certificate Authority เพื่อออก Certificate เพราะค่า URL ของ CRL และ OCSP จะถูกฝังไปใน Certificate ที่ออกไปเพื่อให้ Client สามารถเชื่อมต่อกลับมาสอบถามสถานะการเพิกถอน Certificate ได้และหลังการออก Certificate ไปแล้ว URL ตรงนี้ที่ถูกฝังไว้ใน Certificate ที่ได้สร้างออกไปแล้วจะไม่สามารถเปลี่ยนได้



จากในภาพข้างบน ตัว Server ที่ Run Docker Image นี้จะมี IP Address 123.123.123.123 จึงสามารถสร้าง Base URL ของ CRL และ OCSP ได้ดังนี้

OCSP_BASE_URL=http://123.123.123.123

CRL_BASE_URL=http://123.123.123.123:81

แต่เนื่องจาก URL นี้ใช้ IP Address เป็นส่วนประกอบหากมีการเปลี่ยน IP Address Certificate ทั้งหมดที่ออกจากระบบจะไม่สามารถเช็คได้ว่าถูกเพิกถอนแล้วหรือยังอีกต่อไป จึงแนะนำให้ใช้เป็น Domain Name ซึ่งจะถูกแปลงเป็น IP Address โดยอัตโนมัติด้วยระบบ DNS และมีความสวยงามเข้าใจง่ายกว่าใช้ IP Address ตรงๆ

ในตัวอย่างจะทำการ Config DNS ของ Domain innovative.co.th โดยจะใช้

Sub-Domain ca-poc.innovative.co.th ตั้งค่าที่ผู้ให้บริการ DNS โดยชี้ไปที่ IP Address 123.123.123.123

DNS management for **innovative.co.th**

Review, add, and edit DNS records. Edits will go into effect once saved. [Import and Export](#) [Dashboard Display Settings](#)

Search DNS Records

[Add filter](#) [Search](#) [Add record](#)

ca-poc.innovative.co.th points to **123.123.123.123**.

Type	Name (required)	IPv4 address (required)	Proxy status	TTL
A	ca-poc.innovative.co.th Use @ for root	123.123.123.123	<input checked="" type="checkbox"/> DNS only	Auto

ในตัวอย่างใช้ Cloudflare เป็น DNS Provider โดยหากใช้งาน Cloudflare ให้ปิดการทำงานของระบบ Proxy / CDN ของ Sub-domain นี้ (ปิดเขมส์)

ทดสอบความถูกต้องของ Sub-Domain โดยใช้ Command nslookup ดูผลลัพธ์ของไอพีที่ได้รับจาก DNS Server

```
C:\Users\Administrator>nslookup ca-poc.innovasive.co.th
Server:  dc-dnsresolver.innovasive.co.th
Address: 10.10.100.1

Non-authoritative answer:
Name:    ca-poc.innovasive.co.th
Address: 123.123.123.123

C:\Users\Administrator>_
```

หลังจากตั้งค่า DNS เรียบร้อยจึงสามารถสร้าง Base URL ของ CRL และ OCSP ได้ดังนี้

```
OCSP_BASE_URL=http://ca-poc.innovasive.co.th
```

```
CRL_BASE_URL=http://ca-poc.innovasive.co.th:81
```

โดยให้นำค่าเหล่านี้ไปใส่ไว้ในไฟล์ .env เพื่อไว้ส่งให้ Docker Container ใน Command ตอนรัน Docker Container

```
[chakree@Chakrees-MacBook-Pro ~ % cat .env
OCSP_BASE_URL=http://ca-poc.innovasive.co.th
CRL_BASE_URL=http://ca-poc.innovasive.co.th:81

chakree@Chakrees-MacBook-Pro ~ %
```

- การตั้งค่า Passphrase สำหรับอ่าน Private Key ของ Intermediate Certificate Authority

ในขั้นตอนตามคู่มือของทาง Thai University Consortium จะมีให้สร้าง Private Key ของ Intermediate Certificate Authority โดย OpenSSL จะให้สร้างรหัสผ่านสำหรับเข้ารหัส Private Key (Passphrase) เพื่อความปลอดภัย



2. การสร้าง Intermediate CA

2.1. สร้าง directories และ files ที่จำเป็น

```
$ mkdir univ-ca
$ cd univ-ca
$ mkdir certs crl csr newcerts pkcs12 private
$ chmod 700 private
$ touch index.txt
$ echo 1000 > serial
$ echo 1000 > crlnumber
$ openssl genrsa -aes256 -out private/univ-ca.key.pem 4096
$ chmod 400 private/univ-ca.key.pem
```

** Intermediate CA มีอายุ 10 ปี ควรตั้งรหัสผ่านที่ปลอดภัยสำหรับ private key

โดยหากต้องการใช้งานตัว Docker Image อันนี้เพื่อออก Certificate จำเป็นจะต้องใส่รหัสตัวนี้ให้กับ Docker Image นี้ เพื่อนำไปอ่าน Private Key แล้วทำการ Sign Certificate ให้

โดยทำการระบุใน .env file เช่นหากรหัสผ่านคือ password จะสามารถสร้าง config ได้ดังนี้

```
CA_PRIVKEY_PASSWORD=password
```

โดยให้นำค่าเหล่านี้ไปใส่ไว้ในไฟล์ .env เพื่อไว้ส่งให้ Docker Container ใน Command ตอนรับ Docker Container

```
chakree@Chakrees-MacBook-Pro ~ % cat .env
OCSP_BASE_URL=http://ca-poc.innovasive.co.th
CRL_BASE_URL=http://ca-poc.innovasive.co.th:81
CA_PRIVKEY_PASSWORD=password

chakree@Chakrees-MacBook-Pro ~ %
```

- การตั้งค่า API KEY สำหรับ API ที่ Port 8080 ของ Docker Image

TCP Port 8080 จะเป็น HTTP API Server สำหรับให้ Application หรือ Admin เชื่อมต่อ ผ่านทาง API โดยจะสามารถควบคุม Certificate Authority นี้ได้ผ่านทาง Restful API โดยอ้างอิงการใช้งานได้จากเอกสาร API Documentation ซึ่งจะต้องมีการ Authenticate API ด้วย Static API Key ใน HTTP Request Header (X-INNOVASIVE-API-KEY) ซึ่งจะสามารถตั้งค่า API Key นี้ได้ใน .env File

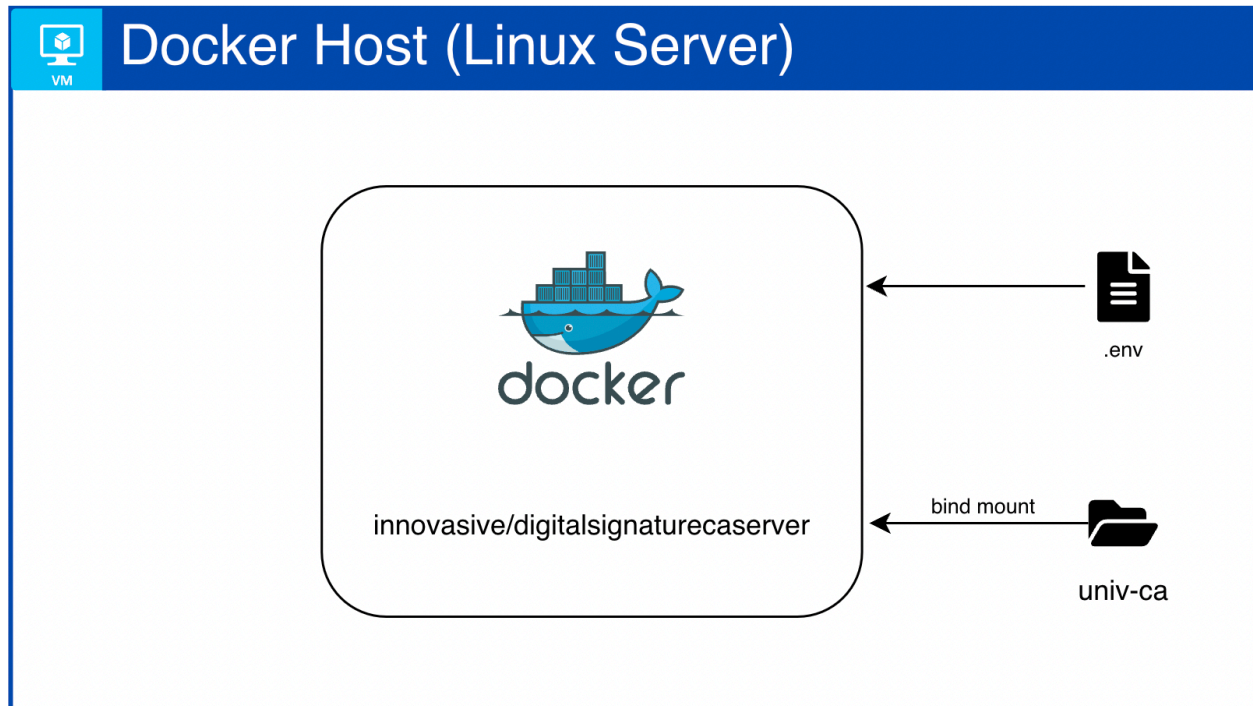
โดยในตัวอย่างจะใช้ API Key developmentkey จะสามารถสร้าง config ได้ดังนี้

```
API_KEY=developmentkey
```

โดยให้นำค่าเหล่านี้ไปใส่ไว้ในไฟล์ .env เพื่อไว้ส่งให้ Docker Container ใน Command ตอนรัน Docker Container

```
chakree@Chakrees-MacBook-Pro ~ % cat .env
OCSP_BASE_URL=http://ca-poc.innovasive.co.th
CRL_BASE_URL=http://ca-poc.innovasive.co.th:81
CA_PRIVKEY_PASSWORD=password
API_KEY=developmentkey
chakree@Chakrees-MacBook-Pro ~ %
```

Start Docker Container Image



--env-file ให้ระบุ Path ของ .env ที่ทำการสร้างมาตามคู่มือนี้ข้างบน

--volume ให้ระบุ Path Folder univ-ca บนเครื่อง เช่นหาก Folder อยู่นอก Folder ปัจจุบัน สามารถใส่เป็น Absolute Path ได้ตัวอย่างเช่น หาก Folder อยู่ที่ /Users/chakree/digitalsignaturecaserver/univ-ca สามารถระบุเป็น

--volume /Users/chakree/digitalsignaturecaserver/univ-ca:/univ-ca

ตัวอย่างนี้จะสามารถใช้ command นี้ run docker image ได้ทันที

```
docker run -d --restart always --name digitalsignaturecaserver \  
--publish 8080:8080 \  
--publish 80:80 \  
--publish 81:81 \  
--env-file .env \  
--volume ./univ-ca:/univ-ca \  
innovative/digitalsignaturecaserver:0.1
```



```

chakree@Chakrees-MacBook-Pro digitalsignaturecaserver %
docker run -d --name digitalsignaturecaserver \
--publish 8080:8080 \
--publish 80:80 \
--publish 81:81 \
--env-file .env \
--volume ./univ-ca:/univ-ca \
innovative/digitalsignaturecaserver:0.1

Unable to find image 'innovative/digitalsignaturecaserver:0.1' locally
0.1: Pulling from innovative/digitalsignaturecaserver
a880266d3b77: Already exists
0e81258012aa: Pull complete
1b1ef9a0c36a: Pull complete
3acd653ed006: Pull complete
fe5f9afdd97c: Pull complete
cd24ffc4b1c: Pull complete
d93ce17932f4: Pull complete
5e31d0bb945c: Pull complete
Digest: sha256:a8880f4484391ec094e6ab293a348ab3ac1545f11ce553f058ee82058ce797c6
Status: Downloaded newer image for innovative/digitalsignaturecaserver:0.1
33a7bf05d7264b43c15c570f77f2b40cf4dd9cddca4051d0ad7d1b7cc662a403
chakree@Chakrees-MacBook-Pro digitalsignaturecaserver %

```

หลังจาก Start Docker Image เรียบร้อยสามารถใช้คำสั่ง

Docker ps

เพื่อดูสถานะ Container ได้ โดย Container ควรจะเป็นสถานะ Up

```

chakree@Chakrees-MacBook-Pro digitalsignaturecaserver % docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                               NAMES
33a7bf05d726   innovative/digitalsignaturecaserver:0.1  "sh /app/openssl_sys..." About a minute ago Up About a minute (healthy) 0.0.0.0:80-81->80-81/tcp, 0.0.0.0:8080->8080/tcp digitalsignaturecaserver
chakree@Chakrees-MacBook-Pro digitalsignaturecaserver %

```

สามารถดู Log ได้โดยใช้คำสั่ง

docker logs digitalsignaturecaserver

```
chakree@Chakrees-MacBook-Pro univ-ca % docker logs digitalsignaturecaserver
```

```

-+*****-.:=+++=-+*****+:.:=+++=-.
*#####. .+++++*****+ -*****+.
*#####+ =+++++*= -*****-
:#####+ -+++++. :+*****-
-#####= =*+++++; .+*****=
=#####- =*****+-----+*****+
+#####: +*****+:+++++= *****+.
*#####. +*****+ .+++++= *****+.
*#####. *****+ .+++++= *****+.
*#####+ :*****= =+++++. :+*****-
:#####= -*****- =+++++. :+*****=
-#####- =*****: -+++++; +*****=
*#####. ***** ++++++ +*****+
*#####* :*****= =+++++ =*****+.
*#####* +*****+ .+++++= *****+.
+#####* +*****+ .+++++= *****+.
=#####: +***** :+++++- *****+.
-#####- =*****: -+++++; *****+.
:#####+ -*****=+++++; ---: *****=
*#####* :****++++++. :++: -*****=
*#####* :*+++++= =---. ....:+++++-
*#####* :+++++ =*****+.
*#####* :+++++* =*****+.
+#####* :+++++* =*****+.
=#####- -+++++*****: .+*****=
.-----.:-----:-----:-----.
```

Digital Signature CA API Server for POC by Innovasive Co., Ltd.

```
Checking if /univ-ca is mounted correctly
/univ-ca is mounted, OK
/univ-ca/index.txt does not exists, creating empty file...
/univ-ca/crlnumber does not exists, creating file...
/univ-ca/serial does not exists, creating file...
/univ-ca/csr does not exists, creating folder...
/univ-ca/newcerts does not exists, creating folder...
/univ-ca/pkcs12 does not exists, creating folder...
/univ-ca/ocsp does not exists, creating folder...
```

```
-----ENVIRONMENT CONFIG-----
Certificate Revocation List (CRL):
http://ca-poc.innovasive.co.th:81/univ-ca.crl

Online Certificate Status Protocol (OCSP):
http://ca-poc.innovasive.co.th

X-Innovasive-API-KEY:
developmentkey
```

```
-----ENVIRONMENT CONFIG-----
```

```
*****
precheck success, starting container
*****
```

```
2023-07-04 15:55:58,678 INFO Set uid to user 0 succeeded
2023-07-04 15:55:58,679 INFO supervisord started with pid 26
2023-07-04 15:55:59,689 INFO spawned: 'crlcronjob' with pid 27
2023-07-04 15:55:59,701 INFO spawned: 'nginx' with pid 28
2023-07-04 15:55:59,706 INFO spawned: 'ocspserver' with pid 29
2023-07-04 15:55:59,708 INFO spawned: 'php-fpm' with pid 30
STARTING OCSP SERVER
```

```
CRL CRON WILL RUN AGAIN IN 1 HOUR
ocsp: waiting for OCSP client connections...
[04-Jul-2023 15:55:59] NOTICE: fpm is running, pid 30
```

หาก Container สถานะเป็น Extied หรือ Failed ตอนสั่ง docker ps คุณสามารถตรวจสอบ Error ได้จาก Log ของ Container เช่น

```
chakree@Chakrees-MacBook-Pro digitalsignaturecaserver % docker logs digitalsignaturecaserver

      -+*****-. :=++++=-+*****+. :=++++=.
      *#####. .+++++****+ -*****+:
      *#####+ =+++++*= -*****-
      :#####+ -+++++., :*****-
      -#####= *+++++.: +*****=
      =#####- =*****+-----+*****+
      +#####: +*****+:+++++= =*****+.
      *#####* .+*****+ .+++++= =*****+:
      *#####* .*****+ =+++++= -*****+-
      *#####: *****= =+++++. :+*****-
      :#####= -*****- =+++++. :+*****=
      -#####= =*****: -+++++: +*****=
      *#####* ***** .+++++= +*****.
      *#####* :*****= =+++++=. -*****-
      *#####* .*****+ .+++++= -*****+:
      +#####* .+*****+ .+++++= =*****+.
      =#####: +*****. :+++++-
      -#####- =*****: -+++++: *****+.
      :#####+ -*****=+++++: .---. :*****=
      *#####+ .***++++++. :+++. -*****=
      *#####* :*+++++= .==-. . . . . :++++-
      *#####* :+++++. =*****+:
      +#####: :+++++** . =*****+.
      =#####- -+++++*****: .+*****=
      .-----.:-----:-----:-----:

Digital Signature CA API Server for POC by Innovasive Co., Ltd.

Checking if /univ-ca is mounted correctly
/univ-ca is mounted, OK
/univ-ca/certs/univ-ca.cert.pem does not exists, exiting script
/univ-ca/certs/univ-ca.cert.pem is not readable, exiting script

*****
ERROR: /univ-ca/certs/univ-ca.cert.pem does not exists
ERROR: /univ-ca/certs/univ-ca.cert.pem is not readable

*****
prechecks failure, exiting container
*****

chakree@Chakrees-MacBook-Pro digitalsignaturecaserver %
```

ในตัวอย่างนี้ Error คือ

/univ-ca/certs/univ-ca.cert.pem does not exists, exiting script

/univ-ca/certs/univ-ca.cert.pem is not readable, exiting script

ให้ทำการตรวจสอบไฟล์ Certificate ว่ามีอยู่ และถูกต้องเป็นต้น แล้วทำการ Start Container ใหม่

ทดสอบสร้างและใช้งาน Certificate

- ทำการตั้ง API KEY ตามที่ได้ระบุไว้ใน .env file ในที่นี้คือ developmentkey

The screenshot shows the 'Authorization' tab in Postman for a collection named 'CA Server API'. The 'Type' is set to 'API Key'. A warning message states: 'Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. Learn more about variables'. The 'Key' is 'X-INNOVASIVE-API-KEY' and the 'Value' is 'developmentkey'. The 'Add to' dropdown is set to 'Header'.

- ทดสอบสร้าง Certificate ด้วย API issue

The screenshot shows the 'Body' tab in Postman for a 'POST' request to 'CA Server API / Issue'. The request body is in 'x-www-form-urlencoded' format. The response is a JSON object indicating success and providing file paths for the certificate and private key.

Key	Value	Description
<input checked="" type="checkbox"/> cn	Somchai Jaidee	
<input checked="" type="checkbox"/> email	somchai@example.ac.th	
<input checked="" type="checkbox"/> password	test1234	
<input checked="" type="checkbox"/> days	30	

```
1 {
2   "success": true,
3   "code": 200,
4   "serial": "01",
5   "certificate_file": "/univ-ca/certs/01.cert.pem",
6   "private_key_file": "/univ-ca/private/01.key.pem",
7   "pkcs12_file": "/univ-ca/pkcs12/01.p12"
8 }
```

- ทดสอบ List Certificate จะต้องพบ Certificate ที่เพิ่งสร้างใหม่

The screenshot shows a REST client interface for a CA Server API. The request is a GET to `{{ENDPOINT}}/listCertificate`. The response is a 200 OK with 31 ms latency and 624 B of data. The response body is displayed in JSON format, showing a list of certificates. The first certificate is highlighted with a green box.

CA Server API / List Certificates

GET `{{ENDPOINT}}/listCertificate` Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

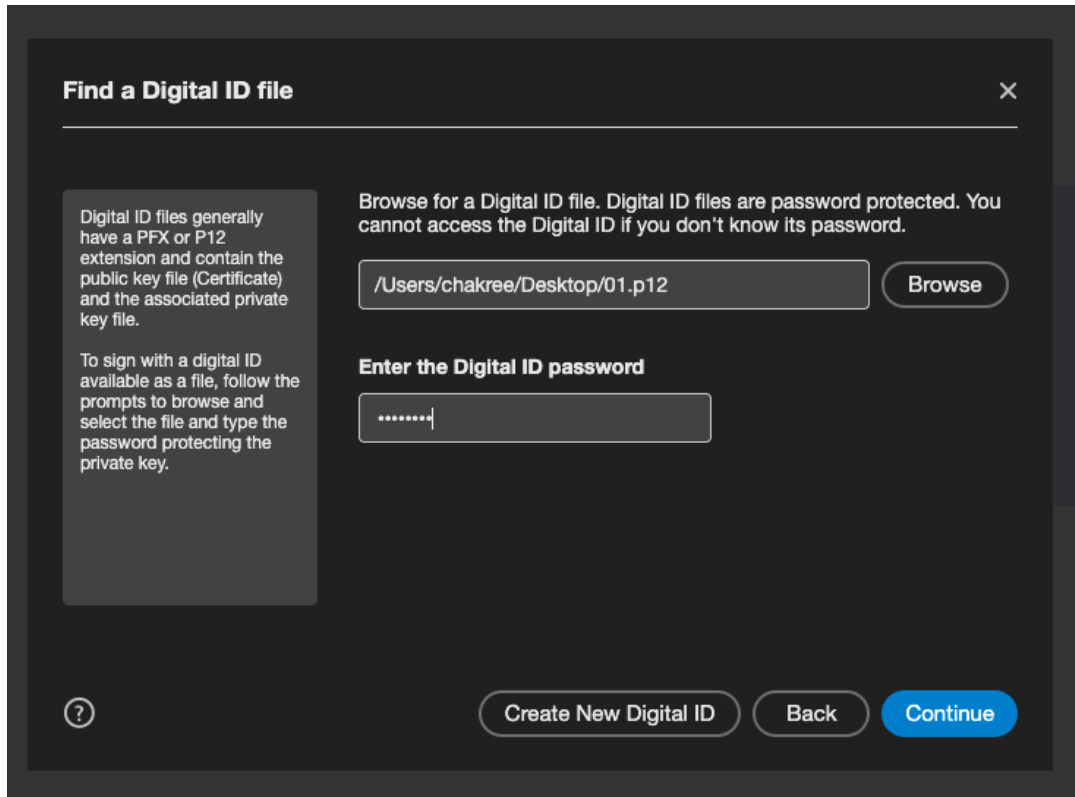
Key	Value	Description	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (5) Test Results 200 OK 31 ms 624 B Save as Example

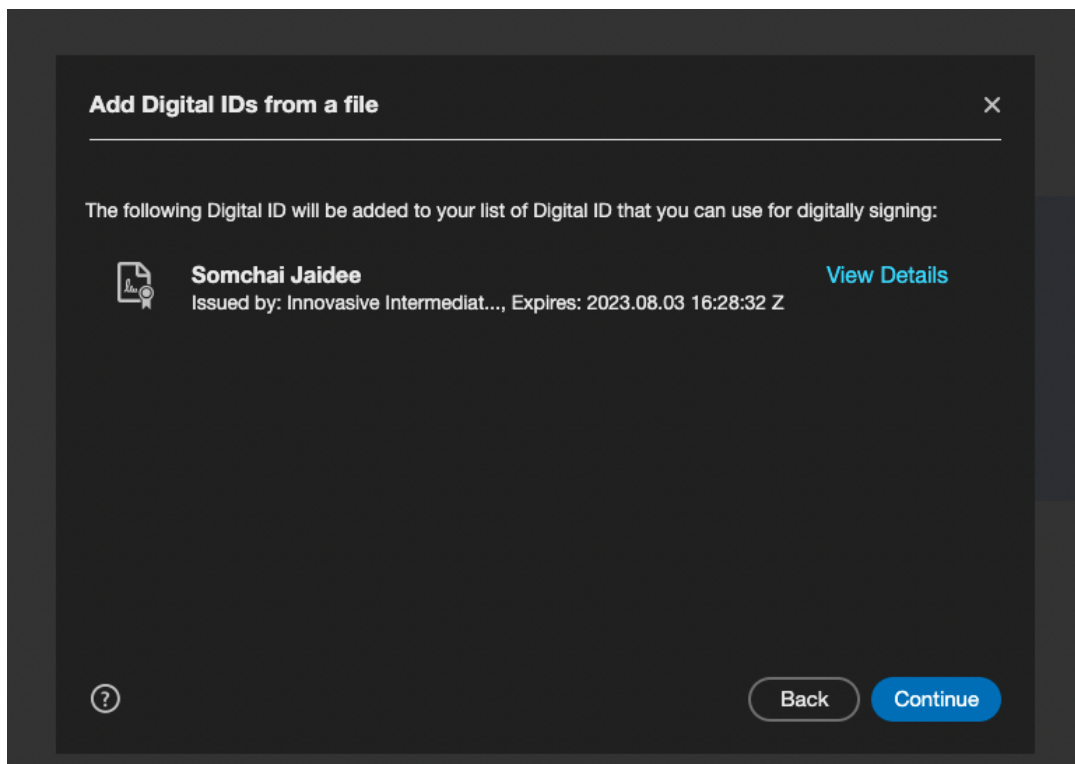
Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "status": "Valid",
4     "expiration_date": "230803160900Z",
5     "revocation_date": "",
6     "serial_number": "01",
7     "subject": "/C=TH/O=Innovasive POC/CN=Somchai Jaidee/emailAddress=somchai@example.ac.th"
8   },
9   {
10    "status": "Valid",
11    "expiration_date": "240429160205Z",
12    "revocation_date": "",
13    "serial_number": "00",
14    "subject": "/C=TH/O=Innovasive POC/CN=OCSP Signer Certificate"
15  }
16 }
```

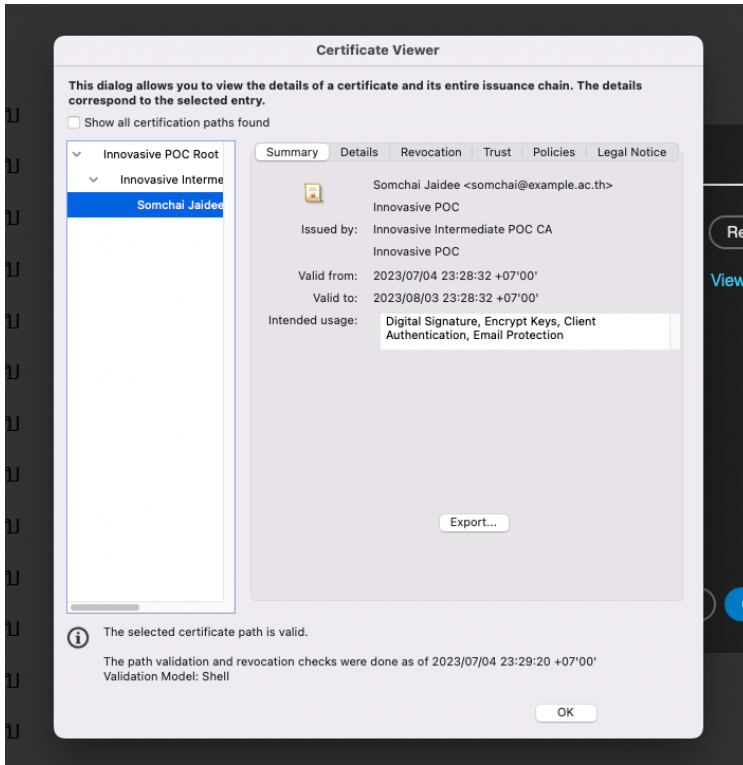
- ทดสอบ นำไฟล์ .p12 ไปใช้งานในโปรแกรมต่อ ในที่นี้จะใช้โปรแกรม Adobe Acrobat ทำการ Sign PDF



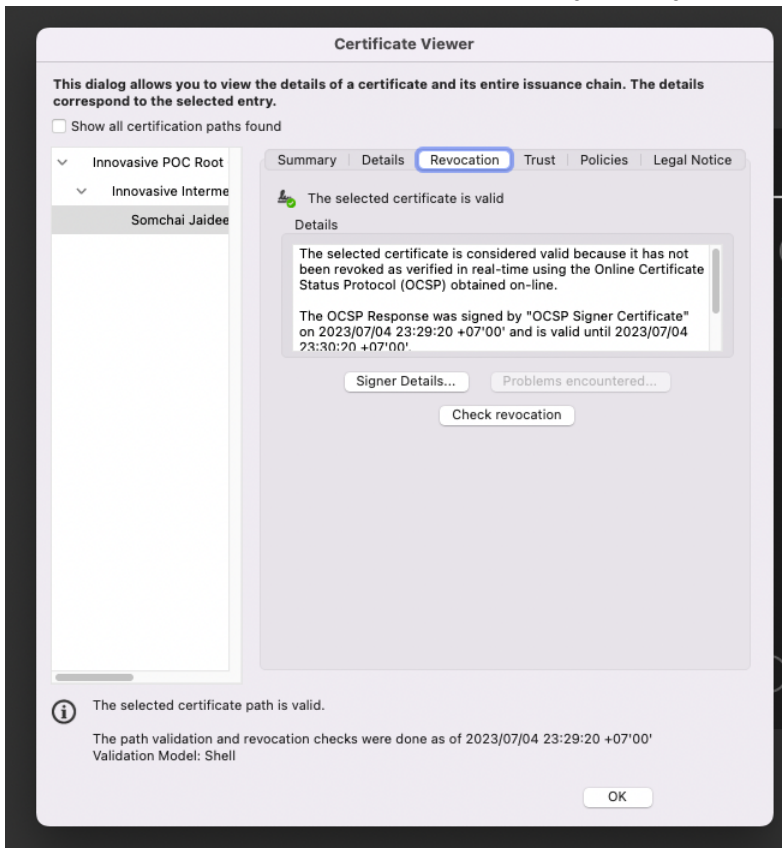
ระบุ Password ที่ใช้ตอนยิง API issue สร้าง Certificate

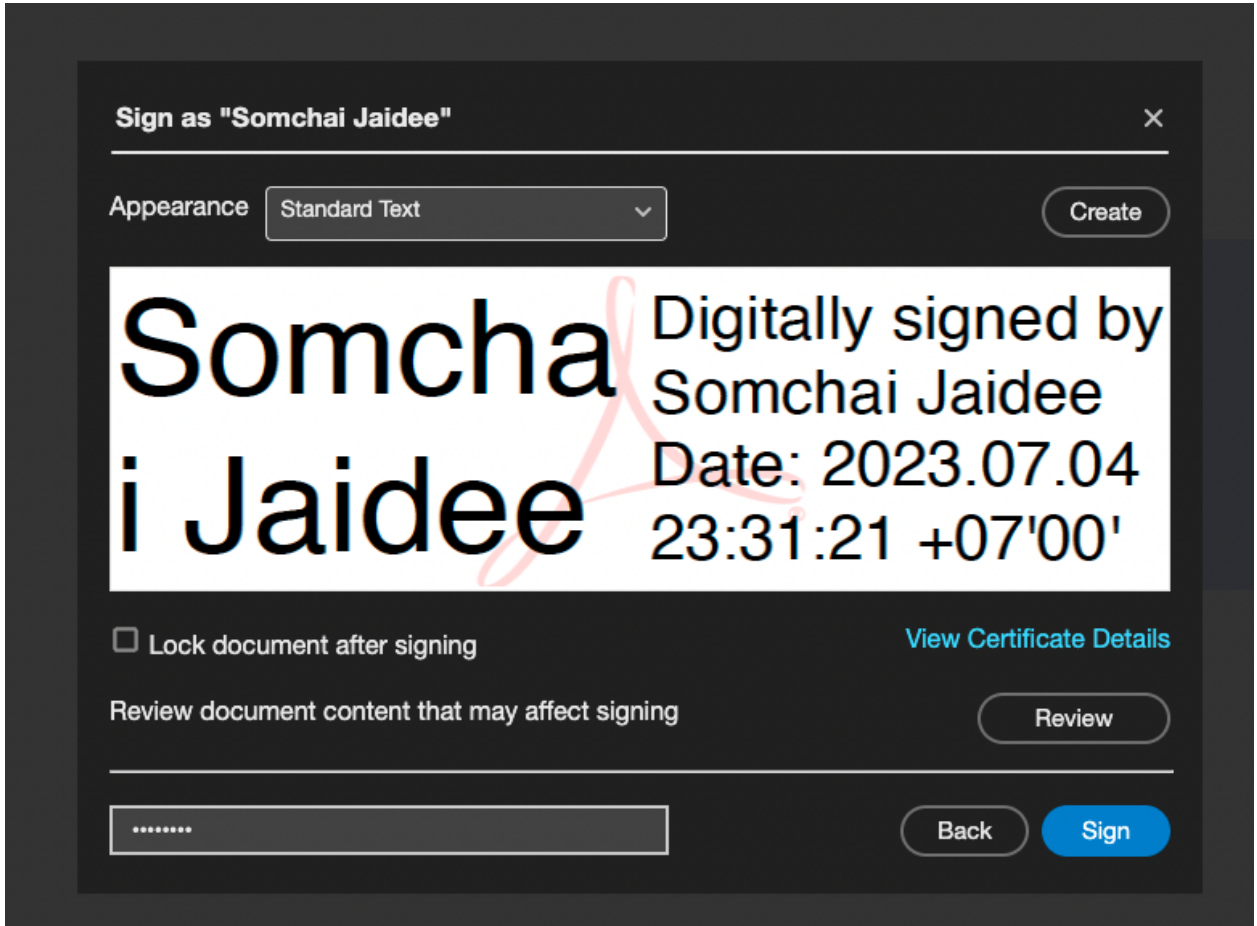


กด View Certificate เพื่อเช็คความถูกต้อง



ตรวจสอบ Revocation จะต้องขึ้นว่าตรวจสอบได้ และข้อมูล Valid ถูกต้อง





ระบุ Password ที่ใช้ตอนยิง API issue สร้าง Certificate

ผลลัพธ์การ Signed PDF จะต้อง Valid ทั้งหมด

The screenshot shows a PDF viewer interface for a document named 'exampledocument_signed.pdf'. The top toolbar includes options for Home, Tools, and Certificates. The Certificates section is active, displaying 'Signed and all signatures are valid.' and a 'Signature Panel' on the right. The main content area shows a list of signatures, all of which are 'Valid'. A 'Signature Properties' dialog box is open, providing detailed information about a specific signature.

Signatures

- Rev. 1: Signed by Somchai Jaidee <somchai@example.ac.th>
Signature is valid:
Source of Trust obtained from manually imported trusted identity.
Document has not been modified since this signature was applied.
Signed by the current user.
Signing time is from the clock on the signer's computer.
Signature is LTV enabled.

Signature Properties

Signature is VALID, signed by Somchai Jaidee <somchai@example.ac.th>.
Signing Time: 2023/07/04 23:31:56 +07'00'
Source of Trust obtained from manually imported trusted identity.

Validity Summary

- The document has not been modified since this signature was applied.
- The certifier has specified that Form Fill-in, Signing and Commenting are allowed for this document. No other changes are permitted.
- The document is signed by the current user.
- Signing time is from the clock on the signer's computer.
- Signature was validated as of the signing time: 2023/07/04 23:31:56 +07'00'

Signer Info

- The path from the signer's certificate to an issuer's certificate was successfully built.
- The signer's certificate is valid and has not been revoked.

Buttons: [Show Signer's Certificate...](#), [Advanced Properties...](#), [Close](#), [Validate Signature](#)

Signature Panel

Digitally signed by Somchai Jaidee
Date: 2023.07.04 23:31:56 +07'00'

ทดสอบ Revoke Certificate

- ทดสอบ Revoke Certificate ด้วย API revoke

The screenshot shows a Postman interface for a REST client. The request is a POST to the endpoint `{{ENDPOINT}}/revoke`. The body is set to `x-www-form-urlencoded`. The request body contains the following data:

Key	Value	Description
serial	01	
cn	Somchai Jaidee	

The response is a 200 OK status with a response time of 62 ms and a body size of 195 B. The response body is shown in JSON format:

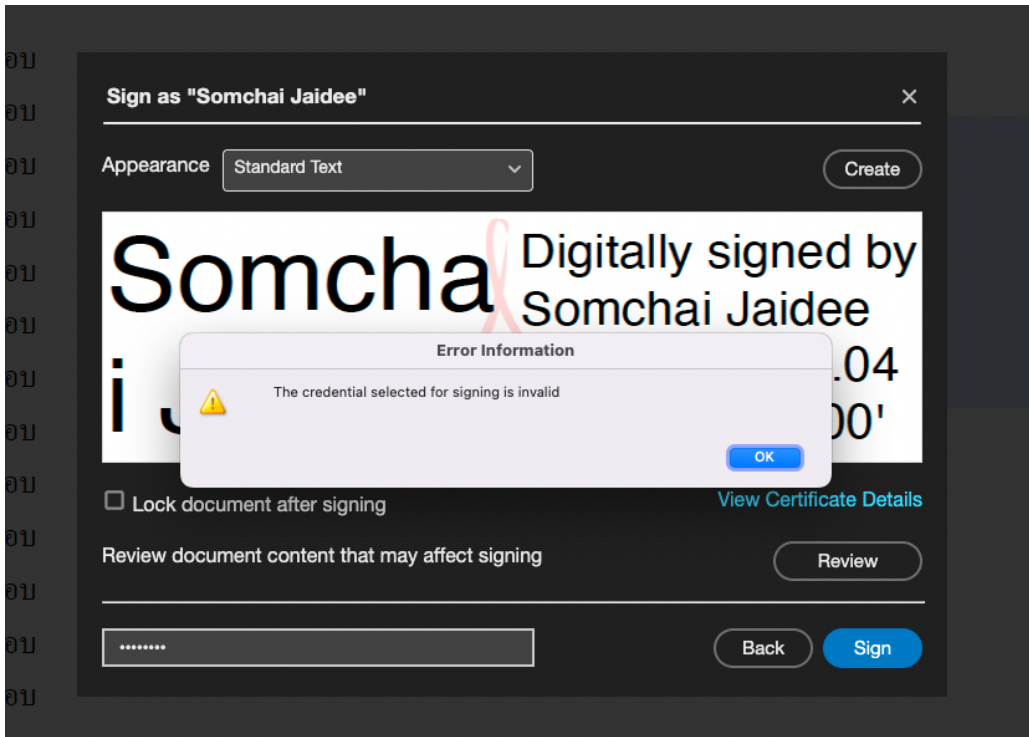
```
1 {
2   "success": true,
3   "code": 200
4 }
```

- ทดสอบ List Certificate จะต้องพบ Certificate ที่เพิ่งถูก Revoke สถานะเป็น Revoked

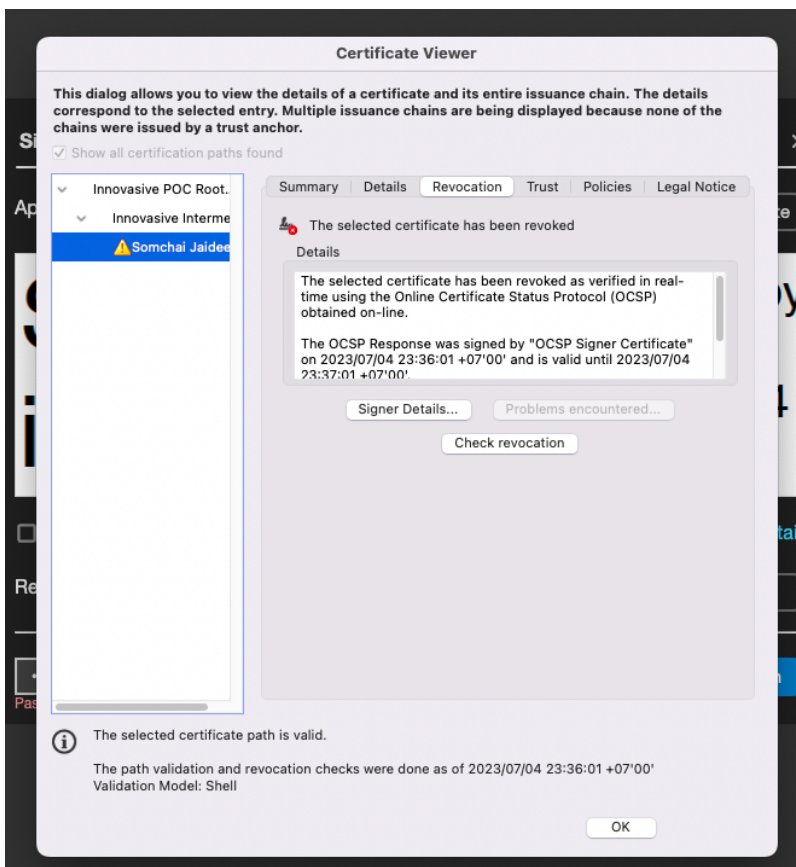
The screenshot shows a Postman interface for a REST client. The request is a GET to the endpoint `{{ENDPOINT}}/listCertificate`. The response is a 200 OK status with a response time of 29 ms and a body size of 650 B. The response body is shown in JSON format:

```
1 {
2   {
3     "status": "Revoked",
4     "expiration_date": "230803162832Z",
5     "revocation_date": "230704163401Z,superseded",
6     "serial_number": "01",
7     "subject": "/C=TH/O=Innovasive POC/CN=Somchai Jaidee/emailAddress=somchai@example.ac.th"
8   },
9   {
10    "status": "Valid",
11    "expiration_date": "240429162817Z",
12    "revocation_date": "",
13    "serial_number": "00",
14    "subject": "/C=TH/O=Innovasive POC/CN=OCSP Signer Certificate"
15  }
16 }
```

- ทดสอบใช้ Certificate ที่ Revoke แล้วทำการ Sign PDF จะพบว่าไม่สามารถ Sign PDF ได้อีกแล้ว



เมื่อตรวจสอบสถานะ: View Certificate Details ใน Tab Revocation จะพบว่า สถานะ Certificate ได้ถูก Revoke แล้ว



สิ้นสุดการทดสอบใช้งาน หากทดสอบตามนี้ได้ผลลัพธ์ถูกต้องทั้งหมด สามารถเริ่มใช้งานได้ทันที