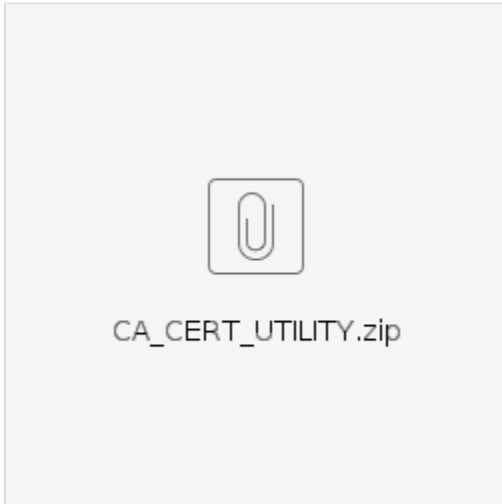


Device Certificate Upload Steps



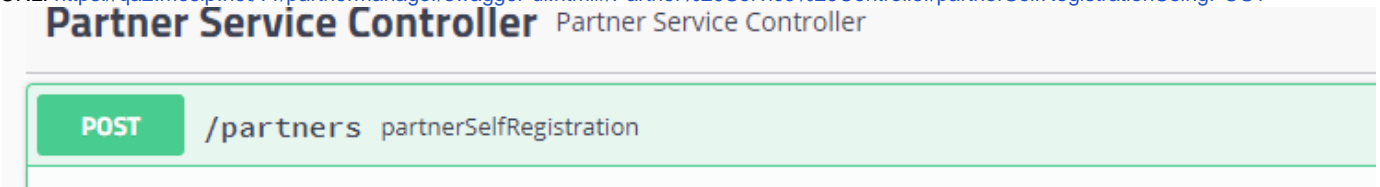
The attachment here CA_CERT_UTILITY is a certificate creation utility that uses shell script commands being executed sequentially to generate valid certificates. For linux machine running the script is easy but for windows machine will need the git being installed or need the openssl application installed in the machine.

Below are the steps for execution:

Note: Change the environment as per the server running in the mentioned swagger link where ever accessing the swagger is accessed.

Create a partner:

URL: <https://qa2.mosip.net/v1/partnermanager/swagger-ui.html#/Partner%20Service%20Controller/partnerSelfRegistrationUsingPOST>



Below is the example:

```
{
  "id": "string",
  "metadata": {},
  "request": {
    "address": "Gandhi marg",
    "contactNumber": "9999999999",
    "emailId": "xyz@gharku.com",
    "organizationName": "DP2",
    "partnerId": "DP2",
    "partnerType": "Device_Provider"
  },
  "requesttime": "",
  "version": "string"
}
```

STEPS To create Certificates:

-Run the "create-certs.sh"

-Sequentially create the certificates for CA, SUBCA and Partner(also known as client) certificates,

1. CA

```

Country Name (2 letter code) [AU]:IN
State or Province Name (full name) [Some-State]:Kar
Locality Name (eg, city) []:Blr
Organization Name (eg, company) [Internet Widgits Pty Ltd]:CA
Organizational Unit Name (eg, section) []:CA
Common Name (e.g. server FQDN or YOUR name) []:CA
Email Address []:
----- Creating Intermediate CA certificate

```

2. SUBCA

```
subject=C = IN, ST = Kar, L = Blr, O = SUBCA, OU = SUBCA, CN = SUBCA
```

3. Partner (Note: the name passed in the partner creation has to be used to create client certificate)

```

Country Name (2 letter code) [AU]:IN
State or Province Name (full name) [Some-State]:Kar
Locality Name (eg, city) []:Blr
Organization Name (eg, company) [Internet Widgits Pty Ltd]:DP2
Organizational Unit Name (eg, section) []:DP2
Common Name (e.g. server FQDN or YOUR name) []:DP2
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Signature ok
subject=C = IN, ST = Kar, L = Blr, O = DP2, OU = DP2, CN = DP2

```

With the completion of the above steps, the certificates are created in the same repository. The required certificate sheets are highlighted below.

Name	Date modified	Type	Size
Client.crt	10-05-2021 13:08	Security Certificate	3 KB
Client.csr	10-05-2021 13:08	CSR File	2 KB
Client.key	10-05-2021 13:06	KEY File	4 KB
Client.key.pkcs8	10-05-2021 13:08	PKCS8 File	4 KB
create-certs.sh	06-05-2021 13:33	Shell Script	2 KB
create-device-keystore.sh	06-05-2021 13:40	Shell Script	1 KB
IntermediateCA.crt	10-05-2021 13:06	Security Certificate	2 KB
IntermediateCA.csr	10-05-2021 13:06	CSR File	2 KB
IntermediateCA.key	10-05-2021 13:03	KEY File	4 KB
IntermediateCA.key.pkcs8	10-05-2021 13:06	PKCS8 File	4 KB
openssl.cnf	06-05-2021 10:08	CONF File	2 KB
README.docx	10-05-2021 13:10	Microsoft Word D...	0 KB
RootCA.crt	10-05-2021 13:03	Security Certificate	2 KB
RootCA.key	10-05-2021 13:02	KEY File	4 KB
RootCA.key.pkcs8	10-05-2021 13:03	PKCS8 File	4 KB

Open the .crt files in "notepad++" to see the certificate file.

Note: Above flow is just a mimic of real scenario, in our system we will use the existing MOSIP certificates as below.

Manual Insertion of the Root(CA cert) and SubRoot(PMS certificate)

Manual insertion is required as of now as these certificates needs to be present in the master.ca_cert_store table.

URL: <https://qa2.mosip.net/v1/keymanager/swagger-ui.html#/keymanager>

For the Root: use "ROOT"

The screenshot shows the Swagger UI for the `/getCertificate` endpoint. The endpoint is a `GET` request. The parameters section shows two query parameters: `applicationId` (string, required) with a value of `ROOT`, and `referenceId` (string, required) with a value of `referenceId - Reference Id as metadata`. The response content type is set to `*/`.

Example : The response of certificate is input for the below insert request in the master.ca_cert_store table

For the SubRoot : use "PMS"

STEPS to upload certificates:

ROOTCA-> INTERMEDIATECA ->Client

Name	Date modified	Type	Size
Client.crt	10-05-2021 13:08	Security Certificate	3 KB
Client.csr	10-05-2021 13:08	CSR File	2 KB
Client.key	10-05-2021 13:06	KEY File	4 KB
Client.key.pkcs8	10-05-2021 13:08	PKCS8 File	4 KB
create-certs.sh	06-05-2021 13:33	Shell Script	2 KB
create-device-keystore.sh	06-05-2021 13:40	Shell Script	1 KB
IntermediateCA.crt	10-05-2021 13:06	Security Certificate	2 KB
IntermediateCA.csr	10-05-2021 13:06	CSR File	2 KB
IntermediateCA.key	10-05-2021 13:03	KEY File	4 KB
IntermediateCA.key.pkcs8	10-05-2021 13:06	PKCS8 File	4 KB
openssl.cnf	06-05-2021 10:08	CNF File	2 KB
README.docx	10-05-2021 13:10	Microsoft Word D...	0 KB
RootCA.crt	10-05-2021 13:03	Security Certificate	2 KB
RootCA.key	10-05-2021 13:02	KEY File	4 KB
RootCA.key.pkcs8	10-05-2021 13:03	PKCS8 File	4 KB

CA Certificates:

URL: <https://qa2.mosip.net/v1/partnermanager/swagger-ui.html#/Partner%20Service%20Controller/uploadCACertificateUsingPOST>

Request:

```
{
  "id": "string",
  "metadata": {},
  "request": {
    "certificateData": "string",
    "partnerDomain": "DEVICE"
  },
  "requesttime": "",
  "version": "string"
}
```

Partner certificate is uploaded as below example:

URL: https://qa2.mosip.net/v1/partnermanager/swagger-ui.html#/Partner%20Service%20Controller/uploadPartnerCertificateUsingPOST_1

Request: assuming partnerID was "DP2"

```
{
  "id": "string",
  "metadata": {},
  "request": {
    "certificateData": "string",
    "partnerDomain": "DEVICE",
    "partnerId": "DP2"
  },
  "requesttime": "",
  "version": "string"
}
```

Note: the certificate is a signed response where the Trust chain has been changed to the MOSIP. This certificate is uploaded in the keymanager DB "Partner_cert_store" table. and then master.ca_cert_store table fetches the partner signed certificate using websub.

The response of the above request is then saved in name “mosip-signed.crt” in the same directory like

[illegible]

```


















# is 65537 (0x010001)
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:IN
State or Province Name (full name) [Some-State]:Kar
Locality Name (eg, city) []:Blr
Organization Name (eg, company) [Internet Wdgts Pty Ltd]:FACE
Organizational Unit Name (eg, section) []:FACE
Common Name (e.g. server FQDN or YOUR name) []:FACE
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:

Signature ok
subject=C = IN, ST = Kar, L = Blr, O = FACE, OU = FACE, CN = FACE
Getting CA Private Key
Enter Export Password:
Verifying - Enter Export Password:

```

Once all Certificates are created, you will find a list of files in the same directory as below

 Client.crt	18-05-2021 11:49	Security Certificate	3 KB
 Client.csr	18-05-2021 11:49	CSR File	2 KB
 Client.key	18-05-2021 11:49	KEY File	4 KB
 Client.key.pkcs8	18-05-2021 11:49	PKCS8 File	4 KB
 create-certs.sh	06-05-2021 13:33	Shell Script	2 KB
 create-device-keystore.sh	06-05-2021 13:40	Shell Script	1 KB
 Device.csr	18-05-2021 12:08	CSR File	2 KB
 Device.key	18-05-2021 12:08	KEY File	4 KB
 Device.p12	18-05-2021 12:08	Personal Informati...	5 KB
 IntermediateCA.crt	18-05-2021 11:49	Security Certificate	3 KB
 IntermediateCA.csr	18-05-2021 11:49	CSR File	2 KB
 IntermediateCA.key	18-05-2021 11:48	KEY File	4 KB
 IntermediateCA.key.pkcs8	18-05-2021 11:49	PKCS8 File	4 KB
 mosip-signed-client.crt	18-05-2021 12:08	Security Certificate	2 KB
 openssl.cnf	06-05-2021 10:08	CNF File	2 KB
 RootCA.crt	18-05-2021 11:48	Security Certificate	3 KB
 RootCA.key	18-05-2021 11:47	KEY File	4 KB

RootCA.key.pkcs8	18-05-2021 11:48	PKCS8 File	4 KB
signed-Device.crt	18-05-2021 12:08	Security Certificate	3 KB

MOCK MDS- build

Download the latest mock MDS .zip from below URL:

<https://github.com/mosip/mosip-mock-services/tree/dev>

The highlighted certificates are then placed in the certificate paths as highlighted in the mock MDS:

Mockmds > qa2-mock > mosip-mock-services-dev > mosip-mock-services-dev > MockMDS > Biometric Devices > Face > Keys

Name	Date modified	Type	Size
Device.p12	17-05-2021 19:12	Personal Informati...	5 KB
signed-Device.crt	17-05-2021 19:10	Security Certificate	3 KB

Update the application.property file as below after placing the certificates as per below path.

s PC > New Volume (D:) > MOSIP > Mockmds > qa2-mock > mosip-mock-services-dev > mosip-mock-services-dev > MockMDS				
Name	Date modified	Type	Size	
Biometric Devices	17-05-2021 12:53	File folder		
files	17-05-2021 12:53	File folder		
Profile	17-05-2021 12:53	File folder		
src	17-05-2021 12:53	File folder		
.gitignore	17-05-2021 12:53	Text Document	1 KB	
application.properties	17-05-2021 20:59	PROPERTIES File	6 KB	
mvnw	17-05-2021 12:53	File	10 KB	
mvnw.cmd	17-05-2021 12:53	Windows Comma...	7 KB	
pom.xml	17-05-2021 12:53	XML Document	8 KB	
README.md	17-05-2021 12:53	MD File	1 KB	
run.bat	17-05-2021 12:53	Windows Batch File	1 KB	
run.sh	17-05-2021 12:53	Shell Script	1 KB	

below is for reference what all are needed to be changed before building the Mock MDS.

```

21 mosip-mock-sbs.biometric.subtype.iris.double.double
22
23 mosip-mock-sbs.file.face.digitalid.json~/Biometric Devices/Face/DigitalId.json
24 mosip-mock-sbs.file.face.deviceinfo.json~/Biometric Devices/Face/DeviceInfo.json
25 mosip-mock-sbs.file.face.deviceDiscovery.json~/Biometric Devices/Face/DeviceDiscovery.json
26 mosip-mock-sbs.file.face.streamImage~/Biometric Devices/Face/Stream Image/2.jpg
27 mosip-mock-sbs.file.face.keys.keystoreFilename~/Biometric Devices/Face/Keys/Device.p12
28 mosip-mock-sbs.file.face.keys.keystorePassword~/Biometric Devices/Face/Keys/signed-Device.crt
29 mosip-mock-sbs.file.face.keys.encryption~/Biometric Devices/Face/Keys/signed-Device.crt
30
31 mosip-mock-sbs.file.finger.slap.digitalid.json~/Biometric Devices/Finger/Slap/DigitalId.json
32 mosip-mock-sbs.file.finger.slap.deviceinfo.json~/Biometric Devices/Finger/Slap/DeviceInfo.json
33 mosip-mock-sbs.file.finger.slap.deviceDiscovery.json~/Biometric Devices/Finger/Slap/DeviceDiscovery.json
34 mosip-mock-sbs.file.finger.slap.streamImage.left~/Biometric Devices/Finger/Slap/Stream Image/2.jpg
35 mosip-mock-sbs.file.finger.slap.streamImage.right~/Biometric Devices/Finger/Slap/Stream Image/2.jpg
36 mosip-mock-sbs.file.finger.slap.streamImage.thumb~/Biometric Devices/Finger/Slap/Stream Image/3.jpg
37 mosip-mock-sbs.file.finger.slap.keys.keystoreFilename~/Biometric Devices/Finger/Slap/Keys/Device.p12
38 mosip-mock-sbs.file.finger.slap.keys.keystorePassword~/Biometric Devices/Finger/Slap/Keys/signed-Device.crt
39 mosip-mock-sbs.file.finger.slap.keys.encryption~/Biometric Devices/Finger/Slap/Keys/signed-Device.crt
40
41 mosip-mock-sbs.file.iris.double.digitalid.json~/Biometric Devices/Iris/Double/DigitalId.json
42 mosip-mock-sbs.file.iris.double.deviceinfo.json~/Biometric Devices/Iris/Double/DeviceInfo.json
43 mosip-mock-sbs.file.iris.double.deviceDiscovery.json~/Biometric Devices/Iris/Double/DeviceDiscovery.json
44 mosip-mock-sbs.file.iris.double.streamImage.left~/Biometric Devices/Iris/Double/Stream Image/2.jpg
45 mosip-mock-sbs.file.iris.double.streamImage.right~/Biometric Devices/Iris/Double/Stream Image/3.jpg
46 mosip-mock-sbs.file.iris.double.keys.keystoreFilename~/Biometric Devices/Iris/Double/Keys/Device.p12
47 mosip-mock-sbs.file.iris.double.keys.keystorePassword~/Biometric Devices/Iris/Double/Keys/signed-Device.crt
48 mosip-mock-sbs.file.iris.double.keys.encryption~/Biometric Devices/Iris/Double/Keys/signed-Device.crt
49
50 mosip-mock-sbs.folder.profile~/Profile
51 mosip-mock-sbs.file.folder.default~/Profile/Default
52

```

Build the MDS in command prompt in the same directory where the pom file exists (run "mvn clean install")