Openssl Cheat Sheet

Resources

- Citrix Documentation for APNS
- Digital Ocean OpenSSL Documentation
- DER v PEM v CRT v CER v KEY

Command Examples

Add locality and other info without interacting with openssl

- I was not able to get this working but you mileage may very.

```
-subj “/C=US/\
        ST=Florida/\
        L=Palm_Harbor/\
        O=Skynet_Healthcare/\
        CN=wso2sso.skynethealthcare.com”

        WSO2Pilot@skynethealthcare
```

Creation

Generate a Private Key & CSR

- Enter a challenge password when prompted

`openssl req -nodes -newkey rsa:2048 -keyout customer.key.pem -out customer-apns-csr.csr`

CSR from Preexisting Key

`openssl req -key domain.key -new -out domain.csr`

Cert from Existing Private Key

`openssl req \ -key domain.key \ -new \ -x509 -days 365 -out domain.crt`

Self-Signed Cert from Existing Private Key & CSR

`openssl x509 -signkey domain.key -in domain.csr -req -days 365 -out domain.crt`

Create a `.pfx` APNS Cert for iOS & MacOS

- You need the Apple MDM Signing cert to complete this command. Can be obtained through the Apple Dev Program.
- Add a password when prompted. This password will be used when uploading the cert to your MDM solution.

```
openssl pkcs12 \ -export -in MDM_Zenprise_Certificate.pem \ -inkey Customer.key.pem \ -out apns_identity.p12
```

Concatenate Certificate Files

At some point, the server cert, intermediate cert(s), and Root cert may need to be combined together then uploaded to a service. For example the VMware IdM Appliance needs the entire cert chain to be in a single file before being uploaded to the appliance.

- All of the files should be in the `PEM` format for this instruction to work.
- Put all of the certs in a single file. In the following order.
  - Server cert at the top
  - Intermediate cert(s) next
  - Root cert at the bottom
- If you are on Linux/MacOS the following command can be used from a CLI session.

  ```
  cat server.crt intermediate.crt Root.crt > concatenated-crt.pem
  ```

- The contents of the file should look similar to the below.

  ```bash
  -----BEGIN CERTIFICATE-----
  MIIHVDCCBjygAwIBAgIRAIA6AEYmeartAAAAAFDyaSMwDQYJKoZIhvcNAQELBQAw

  ... (Server Certificate)

  SAycAaBorDoSVByhvo9N2BaFRAhLgx4wX5BqTyRD76fiIYF3wokeRIgasRt2ZaIB
  dvoGvxUFQ2I=
  -----END CERTIFICATE-----
  -----BEGIN CERTIFICATE-----
  MIIFDjCCA/agAwIBAgIMDulMwwAAAABR03eFMA0GCSqGSIb3DQEBCwUAMIG+MQsw

  ... (Intermediate Certificate)

  exCdtTix9qrKgWRs6PLigVWXUX/hwidQosk8WwBD9lu51aX8/wdQQGcHsFXwt35u
  Lcw=
  -----END CERTIFICATE-----
  -----BEGIN CERTIFICATE-----
  MIIEPjCCAyagAwIBAgIESlOMKDANBgkqhkiG9w0BAQsFADCBvjELMAkGA1UEBhMC

  ... (Root Certificate)

  VHOkc8KT/1EQrBVUAdj8BbGJoX90g5pJ19xOe4pIb4tF9g==
  -----END CERTIFICATE-----
  ```

Conversions

- PEM format (ASCII Readable) - `.pem` , `.cer` , or `.crt`
- DER -Binary file format

Convert PEM to PFX

```
openssl pkcs12 -export -out certificate.pfx -inkey privateKey.key -in certificate.crt -certfile CACert.crt
```

Convert PEM to DER (Binary)

```
openssl x509 \ -in domain.crt \ -outform der -out domain.der
```

Convert DER (Binary) to PEM

```
openssl x509 \
       -inform PEM \
       -outform PEM
       -in domain.crt.der \
       -out domain.crt.pem
```

Convert CSR to DER Format (Binary format for services like APNS)

```
openssl req \
        -inform pem \
        -outform der \
        -in customer.csr \
        -out customer.der
```

Convert Private Key from DER to PEM

```
openssl rsa \
    -inform DER \
    -outform PEM \
    -in server.key \
    -out server.key.pem
```

Additional Resources

- OpenSSL Conversion Commands

Verification

Verify CSR

```
openssl req \
        -text -noout \
        -verify -in domain.csr
```

Verify that Cert and Private Key Match

```
openssl pkey -in privateKey.key -pubout -outform pem | sha256sum
openssl x509 -in certificate.crt -pubkey -noout -outform pem | sha256sum
openssl req -in CSR.csr -pubkey -noout -outform pem | sha256sum
```

View Certificate Contents

View PEM Encoded Cert

- If you get an "Unable to load cert" error then your are trying to view a DER encoded cert.

  Certificates could be `.pem`, `.cer`, or `.crt`

  `openssl x509 -in cert.pem -text -noout`

OCSP URI for CA

```bash
openssl x509 -in cert_name.pem -noout -text | grep -A 2 “Authority Information Access”

Authority Information Access:
       OCSP - URI:http://ocsp.entrust.net
       CA Issuers - URI:http://aia.entrust.net/l1k-chain256.cer
```

View DER Encoded Cert

`openssl x509 \ -in certificate.der \ -inform der -text -noout`

Decryption

Decrypt Private Key

Encrypted Key

```
--BEGIN RSA PRIVATE KEY--
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-256-CBC,AB8E2B5B2D989271273F6730B6F9C687

...

--END RSA PRIVATE KEY--
```

Most of the type, an encrypted key cannot be used directly. It must be decrypted and then used to create additional cert types or uploaded to a server for use.

`openssl rsa -in <encrypted_key.key> -out plain_text.key`

Signing

Sign with SHA1WithRSA Format

- Apple likes for their CSR's to be signed with this format.

```
openssl sha1 \
        -sign private.key \
        -out data.rsa data.der
```

```
openssl req -out wso2sso.skynethealthcare.com.csr \
    -new -newkey rsa:2048 \
    -nodes -keyout wso2sso.skynethealthcare.com.key
```

APNS Cert creation Research

How do sign CSR with MDM signing cert

- CSR to DER format
- Create CSR plist
  - This contains multiple certs
- Submit to Apple APNS portal

Possible option

1. Generate private key:

   `openssl genrsa -out myAppName.key 2048`

2. Generate csr file from private key:

   `openssl req -new -sha256 -key myAppNameCSR.key -out myAppName.csr`

3. Upload csr file to apple to generate certificate

4. Download the certificate

5. Convert certificate to pem file

   `openssl x509 -in aps_development.cer -inform der -out myAppNameCert.pem`

6. Generate pfx file

   `openssl pkcs12 -export -out myAppNameKey.pfx -inkey myAppNameCSR.key -in myAppNa`

7. Convert the pfx file to pem file:

   `openssl pkcs12 -nocerts -out myAppNameKey.pem -in myAppNameKey.pfx`

8. Finally, combine the certificate and key into a single .pem file:

   `cat myAppNameCert.pem myAppNameKey.pem > ckDevelopment.pem`

WSO2 Notes

- followed this article for WSO2 EMM IOS agent APNS integration click here

- This artilce will walk through steps for how to export “in-house” developed iOS app as an enterprise application. Specifically the wso2 emm ios agent app. click here

- Make sure to have your Apple Dev Enterprise Agent reachout to Apple to enable the MDM CSR option in the console. This is one of the filse that you will pass to the python script below.

Say something like ...

> "I am writing to request a MDM Vendor signing certificate for Acme, Inc. I am listed as the Team Agent for our Enterprise Developer Program account."

- I used the following `openssl` commands to generate the customer private key and customer CSR.

  ```
  openssl genrsa -des3 -out [customerPrivateKey].pem 2048
  openssl req -new -key [customerPrivateKey].pem -out [customer].csr
  ```

- Used this script to generate plist file to be uploaded to apple apns sight here
  - `python mdm_vendor_sign.py --csr wso2sso.csr --key wso2ssoPrivateKey.pem --mdm mdm_skynet_healthcare.cer`
  - More notes on the subject here
- Upload this `plist` file to identity.apple.com
-