Self Signed Certificate with Custom Root CA

self-signed-certificate-with-custom-ca.md

Create Root CA (Done once)

Create Root Key

Attention: this is the key used to sign the certificate requests, anyone holding this can sign certificates on your behalf. So keep it in a safe place!

```
openssl genrsa -des3 -out rootCA.key 4096
```

If you want a non password protected key just remove the `-des3` option

Create and self sign the Root Certificate

```
openssl req -x509 -new -nodes -key rootCA.key -sha256 -days 1024 -out rootCA.crt
```

Here we used our root key to create the root certificate that needs to be distributed in all the computers that have to trust us.

Create a certificate (Done for each server)

This procedure needs to be followed for each server/appliance that needs a trusted certificate from our CA

Create the certificate key

```
openssl genrsa -out mydomain.com.key 2048
```

Create the signing (csr)

The certificate signing request is where you specify the details for the certificate you want to generate. This request will be processed by the owner of the Root key (you in this case since you create it earlier) to generate the certificate.

Important: Please mind that while creating the signign request is important to specify the `Common Name` providing the IP address or domain name for the service, otherwise the certificate cannot be verified.

I will describe here two ways to gener

Method A (Interactive)

If you generate the csr in this way, openssl will ask you questions about the certificate to generate like the organization details and the `Common Name` (CN) that is the web address you are creating the certificate for, e.g `mydomain.com` .

```
openssl req -new -key mydomain.com.key -out mydomain.com.csr
```

Method B (One Liner)

This method generates the same output as Method A but it's suitable for use in your automation :) .

```
openssl req -new -sha256 -key mydomain.com.key -subj "/C=US/ST=CA/O=MyOrg, Inc./CN=mydomain.com" -out
mydomain.com.csr
```

If you need to pass additional config you can use the `-config` parameter, here for example I want to add alternative names to my certificate.

```
openssl req -new -sha256 \
    -key mydomain.com.key \
    -subj "/C=US/ST=CA/O=MyOrg, Inc./CN=mydomain.com" \
    -reqexts SAN \
    -config <(cat /etc/ssl/openssl.cnf \
        <(printf "\n[SAN]\nsubjectAltName=DNS:mydomain.com,DNS:www.mydomain.com")) \
    -out mydomain.com.csr
```

Verify the csr's content

```
openssl req -in mydomain.com.csr -noout -text
```

Generate the certificate using the `mydomain` csr and key along with the CA Root key

```
openssl x509 -req -in mydomain.com.csr -CA rootCA.crt -CAkey rootCA.key -CAcreateserial -out
mydomain.com.crt -days 500 -sha256
```

Verify the certificate's content

```
openssl x509 -in mydomain.com.crt -text -noout
```