Manual CA con OpenSSL

La siguiente guía está realizada en un sistema operativo Ubuntu Server 18.04.5

# Archivos necesarios

Para la instalación es necesario contar con los siguientes archivos

* CA.cnf

<https://github.com/nordic-institute/X-Road/blob/develop/ansible/roles/xroad-ca/files/home/ca/CA/CA.cnf>

* sign.sh

<https://github.com/nordic-institute/X-Road/blob/develop/ansible/roles/xroad-ca/files/home/ca/CA/sign.sh>

* revoke.sh

<https://github.com/nordic-institute/X-Road/blob/develop/ansible/roles/xroad-ca/files/home/ca/CA/revoke.sh>

# Comandos necesarios para generar los certificados CA, OCSP y TSA

Los siguientes certificados son necesarios para levantar los servicios CA, OCSP de XACBE y para levantar el Servicio de Sellado de Tiempo (TSA)

## Certificado CA

1. Creamos una carpeta llamada ca dentro del directorio home del servidor con el siguiente comando

*mkdir ca*



1. Nos posicionamos dentro de la carpeta con el comando

*cd ca*

**

1. Dentro de la carpeta ca escribimos el siguiente comando

*rm -rf private certs newcerts crl csr*



1. Luego escribimos el siguiente comando

*rm -f index.\* serial*



1. Escribimos el siguiente comando que nos creara un archivo serial

*echo 01 > serial*



1. Con el siguiente comando creamos los directorios “private”, “certs”, “newcerts”, “crl” y “csr” que servirán para guardar los archivos que genere el CA

*mkdir -p private certs newcerts crl csr*



1. Creamos los archivos index.txt e index.txt.attr con el siguiente comando

*touch index.txt index.txt.attr*



1. Escribimos el siguiente comando

*set -e*



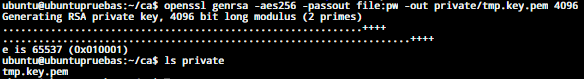
1. Con el siguiente comando creamos el archivo pw donde se almacenará la contraseña que servirá para generar el certificado

*openssl rand -hex -out pw -writerand private/.rnd 16*



1. Generamos una llave temporal con el siguiente comando

*openssl genrsa -aes256 -passout file:pw -out private/tmp.key.pem 4096*



1. El siguiente comando crea la llave para generar el certificado ca

*openssl rsa -in private/tmp.key.pem -passin file:pw -out private/ca.key.pem*



1. Le asignamos los permisos con el siguiente comando

*chmod g+r private/ca.key.pem*



1. Escribimos el siguiente comando que nos generará el certificado ca, donde */O=* corresponde al nombre de la organización y */CN=* corresponde al nombre común. Antes de ejecutar el comando hay que copiar el archivo CA.cnf y todos los de la repo dentro de la carpeta ca

*openssl req -batch -config CA.cnf -subj "/O=CEIQROO/CN=CeiRoot/C=MX" -key private/ca.key.pem -new -x509 -days 7300 -sha256 -extensions v3\_ca -out certs/ca.cert.pem*



**Con estos pasos ya tendremos el certificado ca**

## Certificado OCSP

1. Generamos la llave temporal con el siguiente comando

*openssl genrsa -aes256 -passout file:pw -out private/tmp.key.pem 4096*



1. El siguiente comando crea la llave para generar el certificado ocsp

*openssl rsa -in private/tmp.key.pem -passin file:pw -out private/ocsp.key.pem*





1. Le asignamos los permisos con el siguiente comando

*chmod g+r private/ocsp.key.pem*



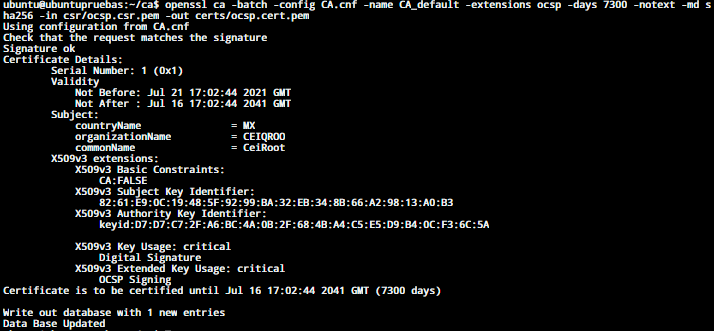
1. Escribimos el siguiente comando que nos generará la petición para el ocsp, donde */O=* corresponde al nombre de la organización y */CN=* corresponde al nombre común.

*openssl req -batch -config CA.cnf -subj "/O=CEIQROO/CN=CeiRoot/C=MX" -key private/ocsp.key.pem -new -sha256 -out csr/ocsp.csr.pem*



1. Con el siguiente comando solicitamos al ca la generación del certificado ocsp

*openssl ca -batch -config CA.cnf -name CA\_default -extensions ocsp -days 7300 -notext -md sha256 -in csr/ocsp.csr.pem -out certs/ocsp.cert.pem*



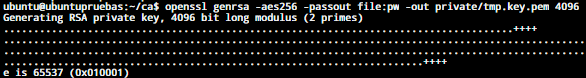
**Con estos pasos ya tendremos el certificado ocsp**



## Certificado TSA

1. Generamos la llave temporal con el siguiente comando

*openssl genrsa -aes256 -passout file:pw -out private/tmp.key.pem 4096*



1. El siguiente comando crea la llave para generar el certificado tsa

*openssl rsa -in private/tmp.key.pem -passin file:pw -out private/tsa.key.pem*



1. Le asignamos los permisos con el siguiente comando

*chmod g+r private/tsa.key.pem*



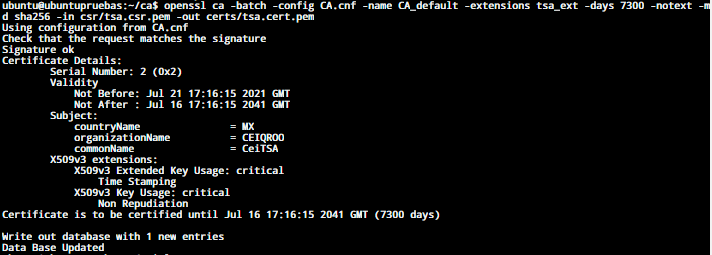
1. Escribimos el siguiente comando que nos generará la petición para el tsa, donde */O=* corresponde al nombre de la organización y */CN=* corresponde al nombre común

*openssl req -batch -config CA.cnf -subj "/O=CEIQROO/CN=CeiTSA/C=MX" -key private/tsa.key.pem -new -sha256 -out csr/tsa.csr.pem*



1. Con el siguiente comando solicitamos al ca la generación del certificado tsa

*openssl ca -batch -config CA.cnf -name CA\_default -extensions tsa\_ext -days 7300 -notext -md sha256 -in csr/tsa.csr.pem -out certs/tsa.cert.pem*



**Con estos pasos ya tendremos el certificado tsa**



Una vez generados los 3 certificados procedemos a eliminar el archivo pw

*rm pw*



# Seccion 2- Sobre las firmas de los security

# Firmar archivos der

Para firmar archivos .der es necesario situarnos dentro de la carpeta ca y copiar esos archivos y ejecutando el siguiente comando con el nombre del archivo nos generará los archivos .pem dentro de la carpeta newcerts/ (Es importante que el archivo sign.sh esté dentro de la carpeta ca)

**Archivo auth:**

*sudo ./sign.sh auth\_csr\_20200625\_securityserver\_QROO\_GOB\_G001\_SSP.der*

**Archivo sign:**

*sudo ./sign.sh sign\_csr\_20200625\_member\_QROO\_GOB\_G001.der*

# Revocar archivos pem

Para revocar los archivos .pem es necesario situarnos dentro de la carpeta ca y ejecutar el siguiente comando pasandole el archivo pem que se quiere revocar (Es importante que el archivo revoke.sh esté dentro de la carpeta ca)

*sudo ./revoke.sh newcerts/34.pem*

# Levantar servicio OCSP

El servicio ocsp es necesario para consultar la valides de los certificados pem. Nos ubicamos dentro de la carpeta ca y con el siguiente comando levantamos ese servicio

*openssl ocsp -index index.txt -port 8888 -rsigner certs/ocsp.cert.pem -rkey private/ocsp.key.pem -CA certs/ca.cert.pem -text -out log.txt -ignore\_err*

**

Comando para exporta p12 del TSA

El servidor de estampado de tiempo requiere un archivo .p12 que servirá para levantar su servicio. Nos ubicamos en la carpeta ca y se exporta con el siguiente comando donde -name es el nombre que le pondremos y nos pedirá una contraseña

*openssl pkcs12 -export -inkey private/tsa.key.pem -in certs/tsa.cert.pem -name CeiTSA -out final\_result.p12*

**