Guide de génération de requête de signature de certificat avec open SSL sous linux

Première étape :

- Installez le package OpenSSL sous linux :
  Pour installer openssl sous ubuntu ou debian, utiliser la commande suivante :
  sudo apt-get install openssl
  Pour installer open ssl sous redhat ou centos utiliser la commande suivante :

```
# yum install openssl
```

Deuxième étape :

- Générez la clé privée en utilisant la commande suivante :

  genrsa -out privatekey.key 2048

privatekey.key : le nom de votre clé, vous pouvez nommer votre clé comme vous voulez.

Remarque : la taille de clé privé doit être 2048 bit.

Exemple :

```
[root@mail abbassi]# openssl genrsa -out privatekey.key 2048
Generating RSA private key, 2048 bit long modulus
.........+++
.....................................................................+++
e is 65537 (0x10001)
[root@mail abbassi]#
```

Remarque importante :

Conservez votre clé privée pour l'utiliser ultérieurement.

Troisième étape : Génération de la requête CSR

Générez la requête de signature de certificat en utilisant la commande suivante **:**

req -new -key privatekey.key -out server.csr -sha256

Private key.key : le nom de la clé privé.

Server.csr : le nom de la requête de signature de certificat (csr), vous pouvez la nommer comme vous voulez.

Le système va vous demander de saisir des champs ; remplissez-les en respectant les instructions suivantes:

Country Name : (2 letter code) [le nom de votre pays]: (TN)
State or Province Name : (full name) [Some-State]: (TUNISIE)
Locality Name : (le nom de votre ville)
Organization Name : (le nom de votre organisation)
Organizational Unit Name (section) []: (le nom de votre service)
Common Name : (le nom du site web a sécuriser)
Email Address : (l'adresse email de l'administrateur de site web)

Remarque :

- l'adresse email doit être postmaster@votredomaine.tn ou admin@votredomain.tn ou webmaster@votredomaine.tn.
- Les caractères suivants ne sont pas acceptés : é è ^ à < > ~ ! @ # $ % ^ * / \ ( ) ?.,&

Exemple :

req –new -key privatekey.key -out server.csr -sha256

```
[root@mail abbassi]# openssl req -new -key privatekey.key -out server.csr -sha256
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) [XX]:TN
State or Province Name (full name) []:TUNISIE
Locality Name (eg, city) [Default City]:TUNIS
Organization Name (eg, company) [Default Company Ltd]:ANCE
Organizational Unit Name (eg, section) []:SERVICE INFORMATIQUE
Common Name (eg, your name or your server's hostname) []:www.mydomaine.tn
Email Address []:webmaster@mydomaine.tn

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

Quatrième étape : vérification de requête

Pour afficher la requête de signature de certificat CSR, utilisez la commande suivante :

req -text -in nom de votre requête CSR

Exemple :

req –text -in server.csr

server.csr : le nom de requête CSR .

```
[root@mail abbassi]# openssl req -text -in server.csr
Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: C=TN, ST=TUNISIE, L=TUNIS, O=ANCE, OU=SERVICE INFORMATIQUE, CN=www.mydomaine.tn/emailAddress=webmaster@mydomaine.tn
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:a8:a2:9c:d1:52:06:d8:5b:5c:5e:f8:46:3e:a5:
                    11:43:23:54:ef:b1:76:7e:7a:b3:17:7a:ea:13:c8:
                    ec:7b:fe:2e:20:94:02:9d:d9:dc:1f:14:89:37:a1:
                    cd:65:b2:f0:74:db:f1:4d:f9:8d:c5:71:63:0a:e6:
                    4a:91:c7:52:e3:28:e1:f2:20:35:4d:8e:53:31:0f:
                    d4:00:7f:ab:2c:6f:a2:4d:14:0b:d4:c8:e8:97:9e:
                    6b:c8:a0:19:44:e9:13:32:5d:8a:34:5b:e6:bc:d3:
                    16:e7:f2:96:91:7e:12:cf:75:aa:3c:49:00:cd:41:
                    20:e7:c5:c2:04:fc:16:5c:7f:e8:ba:d3:4a:4d:8b:
                    e7:b8:98:35:fd:dd:ac:3f:e9:45:19:e3:fd:d4:09:
                    97:ec:c5:b7:d7:e3:e1:f5:f9:bc:82:50:c2:59:98:
                    b0:e4:90:bd:9b:b6:9e:82:a5:fd:c4:8f:ee:3f:b7:
                    65:4a:6c:46:65:fa:3e:ca:23:20:31:81:52:a4:d1:
                    ad:cc:a3:d6:9c:ef:a0:c7:88:e5:fe:f1:ba:69:56:
                    45:e4:55:98:a0:93:8c:99:2a:23:43:f7:78:0f:3f:
                    d0:10:76:34:5a:1d:35:f3:0f:69:ca:f8:df:58:3e:
                    57:4e:a8:f4:ef:88:4b:da:97:2b:7a:e2:ac:3f:c5:
                    a7:73
                Exponent: 65537 (0x10001)
        Attributes:
            a0:00
    Signature Algorithm: sha256WithRSAEncryption
         0e:24:bb:7a:f2:0d:68:b9:6c:2d:9d:80:ff:e2:a3:02:8d:16:
         96:23:58:49:01:4b:3e:dd:71:ec:c8:f1:61:cd:e8:68:49:68:
         21:f8:e0:73:11:a8:e5:56:8d:c1:b6:6b:51:7d:4d:0e:3b:04:
```