Create Self-Signed certificate

First we have to install the package `openssl`

```
apt install openssl
```

Then we are going to create the certificate

```
openssl req -x509 -sha256 -newkey rsa:2048 -keyout certificate.key -out
certificate.crt -days 1024 -nodes
```

Parameters guide

- **-newkey rsa:2048** indica que queremos que la clave se genere usando el algoritmo RSA con 2048 bits.
- **-nodes** indica que no queremos que la clave privada sea encriptada con una contraseña.
- **-keyout** es la ruta + nombre del archivo donde queremos que se escriba la clave.
- **-out** es la ruta + nombre del archivo donde queremos que se escriba el documento CSR.
- **-sha256** es el algoritmo hash que queremos usar para generar la huella digital (si la CA a la que queramos enviar el CSR no soporta sha256 sencillamente eliminar esta línea).
- **-x509** indica que queremos crear un certificado autofirmado.
- **-days** indica el número de días durante los cuales es útil (válido) el certificado. Cuando pasen estos días, el navegador (o la aplicación que usemos para realizar la conexión segura) nos indicará que el certificado ya ha expirado. Tengamos en cuenta que el algoritmo de comunicación por Diffi-Hellman se basa en el problema de calcular la inversa del logaritmo en módulo N que aún no ha sido resuelto matemáticamente hablando, solo puede ser deducido mediante fuerza bruta. Dependiendo de la capacidad de computación actual y el número de bits usado en la clave privada, un usuario malicioso puede tardar más o menos tiempo en deducir la clave privada a partir de la clave pública. Para ello ponemos un límite en el cual suponemos que un usuario malicioso podría haber «roto» la clave y en ese momento el certificado SSL ya no sería seguro.