

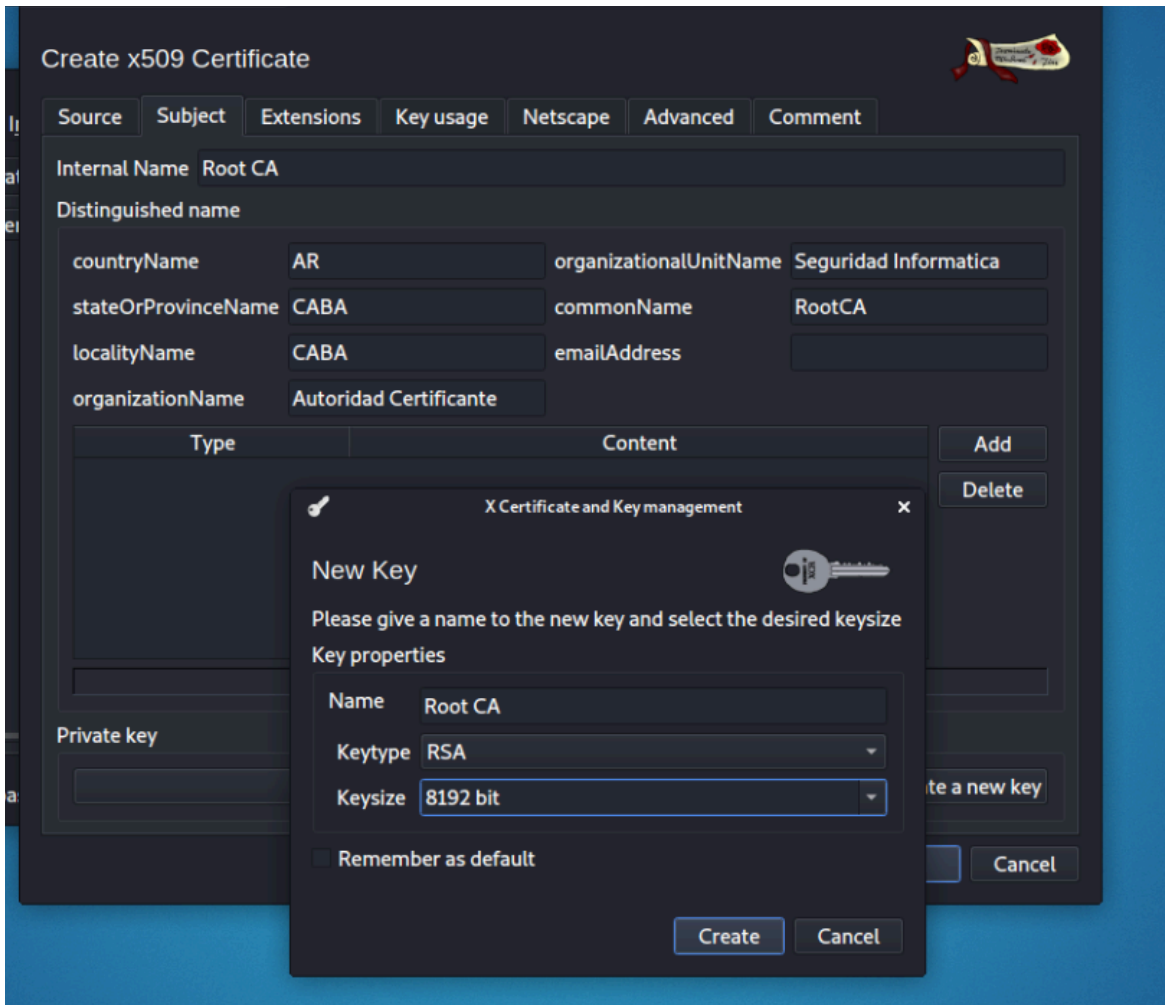
# Lab PKI informe

## Todo el grupo 06

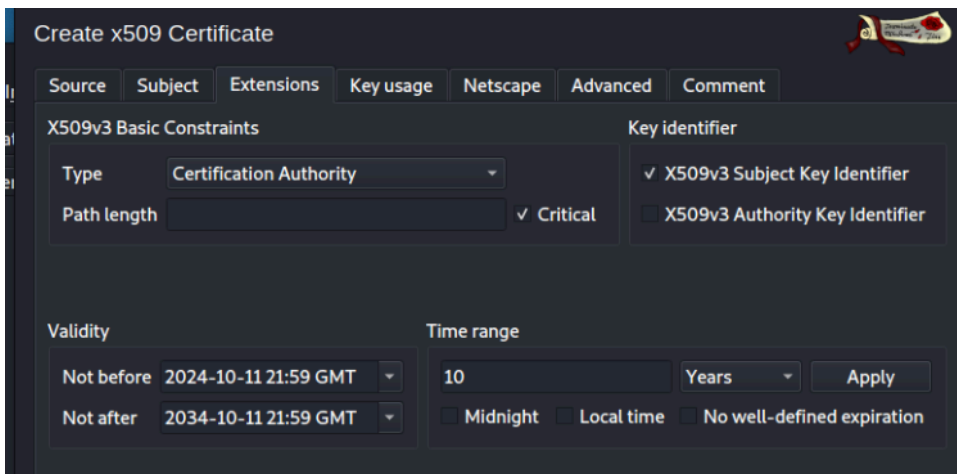
**XCA:** Herramienta gráfica para gestionar nuestros certificados y creación de entidades certificadoras y certificados clientes/servidor.

Procedimiento:

- 1) Se crea la base de datos **xca01** con contraseña **firstxca01**
- 2) Creamos el Root CA cuyo algoritmo de cifrado para la **key privada** sera **RSA** con 8192 bits.

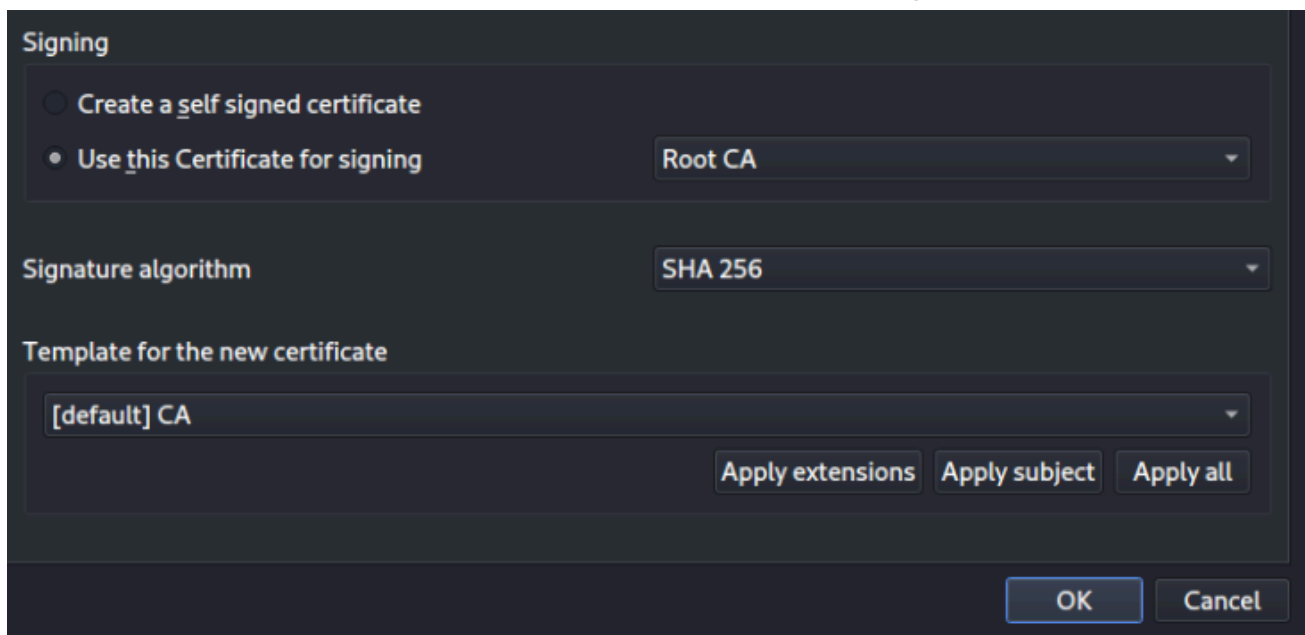


Nos aseguramos que el certificando tenga 10 años de duracion.



## Creación de un SubCA:

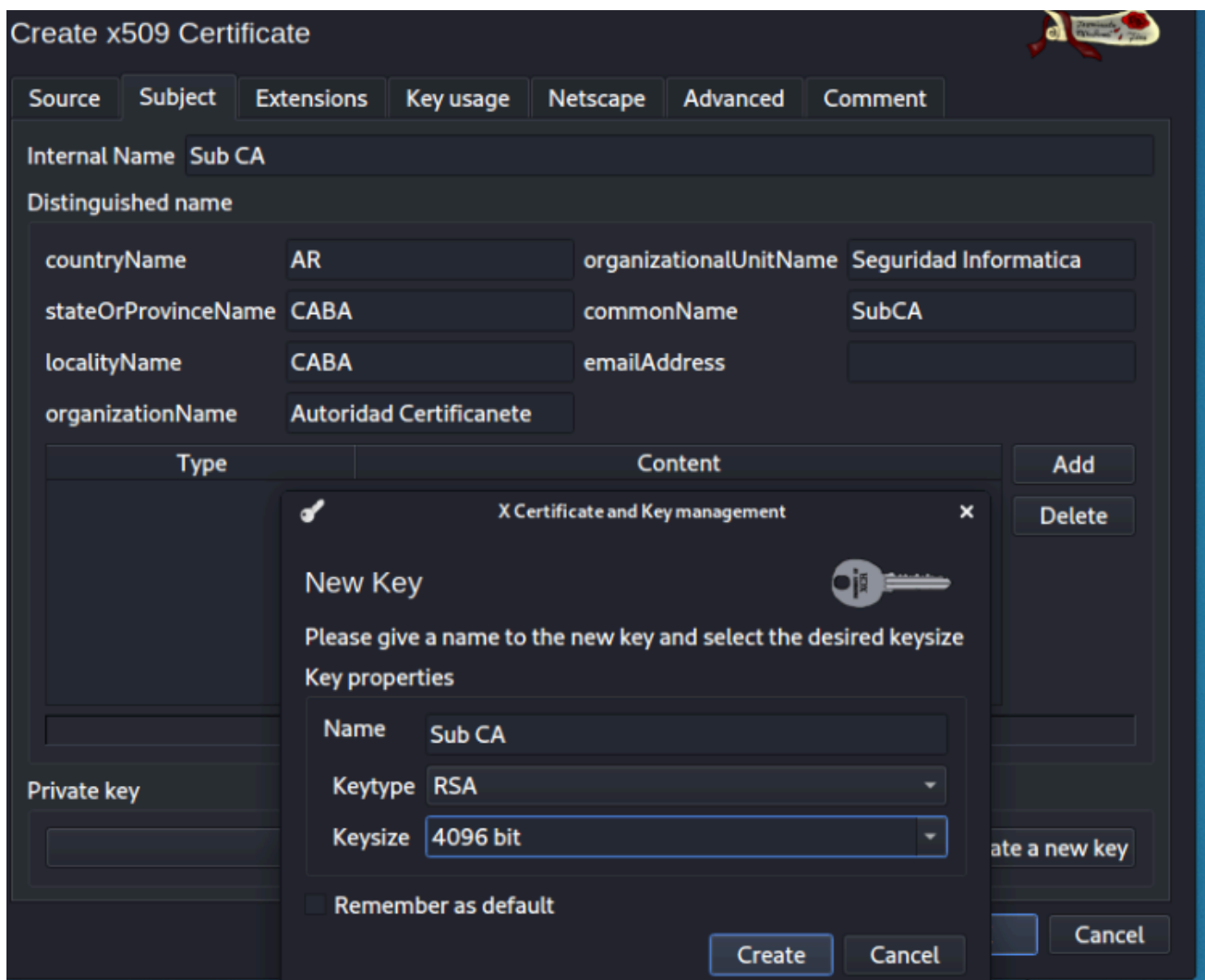
1) Creamos el certificado de la subCA: usando el algoritmo SHA 256.



The 'Signing' dialog box is shown with the following configuration:

- Signing:** ☒ Use this Certificate for signing. The dropdown menu shows 'Root CA'.
- Signature algorithm:** SHA 256.
- Template for the new certificate:** [default] CA.
- Buttons:** 'Apply extensions', 'Apply subject', 'Apply all', 'OK', and 'Cancel'.

2) Completamos los datos y creamos la private key usando el algoritmo RSA 4096 bit.



The 'Create x509 Certificate' dialog box is shown with the 'Subject' tab selected. The 'Internal Name' is 'Sub CA'. The 'Distinguished name' section contains the following fields:

Field	Value
countryName	AR
stateOrProvinceName	CABA
localityName	CABA
organizationName	Autoridad Certificanete
organizationalUnitName	Seguridad Informatica
commonName	SubCA
emailAddress	

The 'Private key' section is visible at the bottom. A 'New Key' dialog box is open over the 'Private key' section, showing the following configuration:

- New Key:** Please give a name to the new key and select the desired keysize.
- Key properties:**
  - Name: Sub CA
  - Keytype: RSA
  - Keysize: 4096 bit
- Remember as default:** ☐
- Buttons:** 'Create', 'Cancel', 'Add', and 'Delete'.

3) Configuramos una duración de 5 años

X Certificate and Key management

## Create x509 Certificate

Source Subject Extensions Key usage Netscape Advanced Comment

**X509v3 Basic Constraints**

Type: Certification Authority

Path length:  ☒ Critical

**Key identifier**

☒ X509v3 Subject Key Identifier

☐ X509v3 Authority Key Identifier

**Validity**

Not before: 2024-10-11 22:22 GMT

Not after: 2029-10-11 22:22 GMT

**Time range**

5 Years

☐ Midnight ☐ Local time ☐ No well-defined expiration

4) Configuramos con éxito la SubCA para firmar los certificados para los web server (Apache, Nginx).

X Certificate and Key management

File Import Token Extra Help

Private Keys Certificate signing requests Certificates Templates Revocation lists

Internal name	commonName	CA	Serial	Expiry date	CRL Expiration
Root CA RootCA	✓ Yes	29430B3822B6D975	10/11/34		
Sub ... SubCA	✓ Yes	43E5486E13B84FAD	10/11/29		

New Certificate

Export

Import

Show Details

Delete

Import PKCS#12

Import PKCS#7

Plain View

5) Generamos la **Certificate Signing Request (CSR)** y una clave RSA para un web server (**apache**).

```

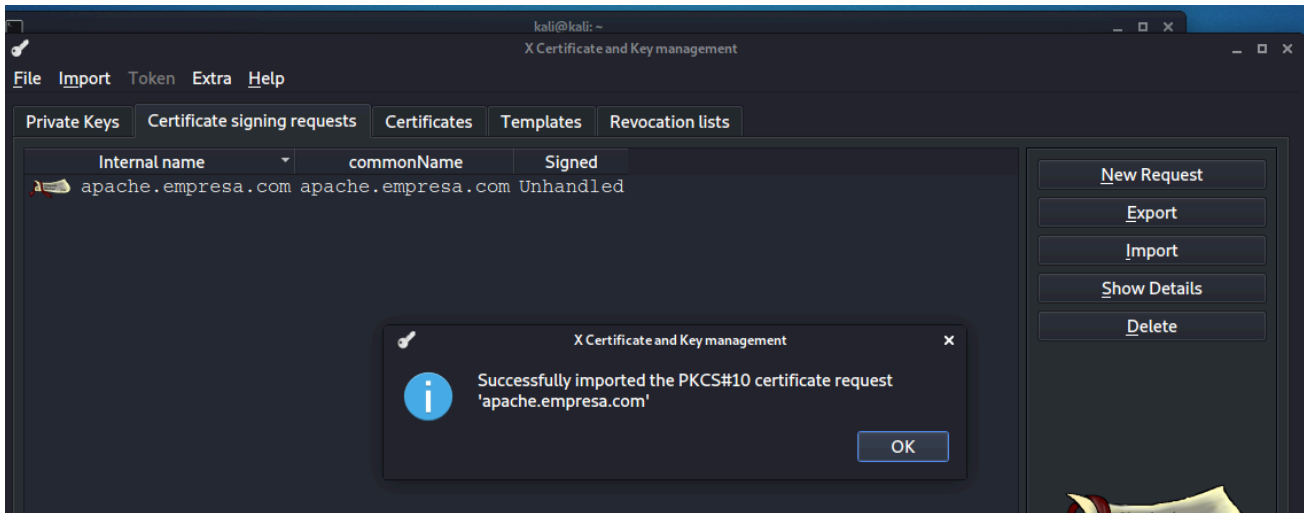
(kali㉿kali)-[~]
$ openssl req -out apache.csr -new -newkey rsa:2048 -nodes -keyout apache.key
Generating a RSA private key
.....+++++CommonName      CA              Serial              Expiry date      CRL Expiration
+++++
writing new private key to 'apache.key'
-----
You are about to be asked to enter 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) [AU]:AR
State or Province Name (full name) [Some-State]:CABA
Locality Name (eg, city) []:CABA
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Empresa
Organizational Unit Name (eg, section) []:Seguridad Informatica
Common Name (e.g. server FQDN or YOUR name) []:apache.empresa.com
Email Address []:

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

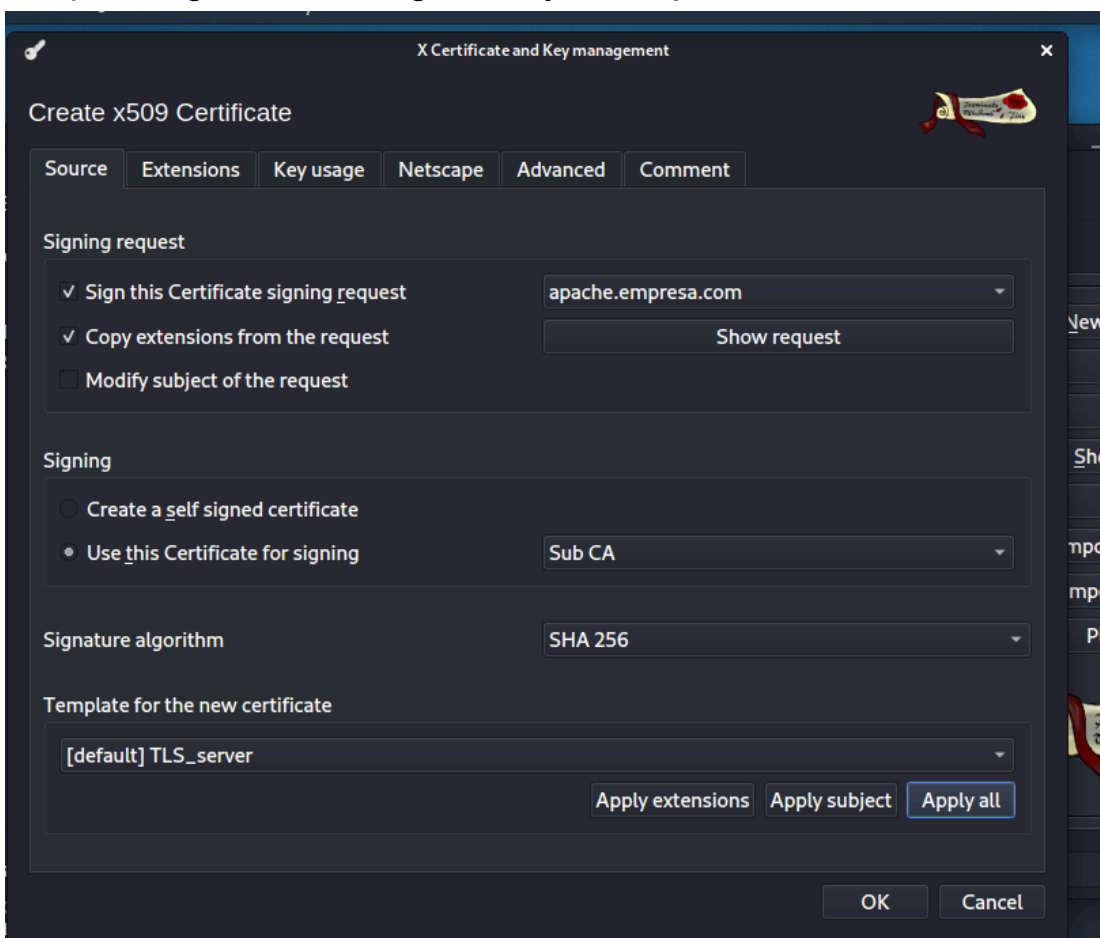
(kali㉿kali)-[~]
$ █

```

## 6) Ahora importamos el **CSR** a la **CA**:



## 7) Configuramos el algoritmo y el csr para firmar:



## 8) Configuramos el rango de tiempo y los DNS:

Source Extensions Key usage Netscape Advanced Comment

Create x509 Certificate

X509v3 Basic Constraints

Type End Entity

Path length Critical

Key identifier

X509v3 Subject Key Identifier

X509v3 Authority Key Identifier

Validity

Not before 2024-10-11 23:11 GMT

Not after 2025-10-11 23:11 GMT

Time range

365 Days

Midnight Local time No well-defined expiration

X509v3 Subject Alternative Name DNS:copycn, DNS:prueba.empresa.com

X509v3 Issuer Alternative Name

X509v3 CRL Distribution Points

Authority Information Access

OCSP Must Staple

OK Cancel

firmando el **CSR** con nuestro **Sub CA**.

## 9) Obteniendo el certificado digital para ser utilizado por un web server ej: apache. **Exportamos el certificado: (.crt)**

File Import Token Extra Help

Private Keys Certificate signing request

Internal name commonName

Root CA RootCA

Sub CA SubCA

apa... apache.e

Certificate export

Name apache.empresa.com

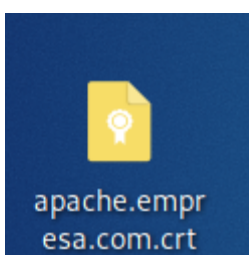
Filename /home/kali/Desktop/apache.empresa.com.crt

PEM Text format with headers

Export comment into PEM file

Export Format PEM (\*.crt)

OK Cancel



- 10) Configuramos el **modulo TLS** y creamos el Virtual host usando **nano y sudo**:

```
(kali@kali)~$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
    systemctl restart apache2
```

```
(kali@kali)~$ nano /etc/apache2/sites-enabled/000-default.conf
GNU nano 5.4 /etc/apache2/sites-enabled/000-default.conf *
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

<VirtualHost *:443>
    ServerName apache.empresa.com
    SSLEngine on
    SSLCertificateFile "/home/kali/Desktop/apache.empresa.com.crt"
    SSLCertificateKeyFile "/home/kali/apache.key"
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

- 11) Ahora actualizamos los dns en /etc/hosts:



```
kali@kali: ~  
File Actions Edit View Help  
  
(kali@kali)-[~]  
$ cat /etc/hosts  
127.0.0.1 localhost  
127.0.1.1 kali  
10.185.3.174 apache.empresa.com  
  
# The following lines are desirable for IPv6 capable hosts  
::1 localhost ip6-localhost ip6-loopback  
ff02::1 ip6-allnodes  
ff02::2 ip6-allrouters  
  
(kali@kali)-[~]  
$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000  
    link/ether 08:00:27:6e:51:16 brd ff:ff:ff:ff:ff:ff  
    inet 172.20.0.72/16 brd 172.20.255.255 scope global dynamic noprefixroute eth0  
        valid_lft 1860sec preferred_lft 1860sec  
    inet6 fe80::a00:27ff:fe6e:5116/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
  
(kali@kali)-[~]  
$ cat /etc/hosts  
127.0.0.1 localhost  
127.0.1.1 kali  
172.20.0.72 apache.empresa.com  
  
# The following lines are desirable for IPv6 capable hosts  
::1 localhost ip6-localhost ip6-loopback  
ff02::1 ip6-allnodes  
ff02::2 ip6-allrouters  
  
(kali@kali)-[~]  
$
```

Le ponemos la ip de nuestra propia maquina (vemos usando ip a)  
(172.20.0.72)

12) Finalmente entramos a la pagina:

