



## Tabla de contenido

1. Introducción .....	3
2. Generación del certificado .....	4
2.1. Instalación de repositorios necesarios .....	4
2.2. Configuración del certificado .....	4
3. Configuración del servidor FTP .....	6
3.1. Habilitar el módulo TLS .....	6
4. Pruebas realizadas .....	7
4.1. Conexión .....	7
5. Bibliografía.....	8

## 1. INTRODUCCIÓN

---

Partiendo de la práctica anterior, vamos a configurar el servidor para que a partir de ahora las comunicaciones las realice a través de un protocolo seguro, como es SFTP que cifra las comunicaciones con un certificado y encripta la comunicación de modo que, si se filtra alguna comunicación, esta no vaya en texto plano como ocurría con la práctica anterior.

## 2. GENERACIÓN DEL CERTIFICADO

### 2.1. Instalación de repositorios necesarios

Lo primero que tenemos que hacer es instalar el repositorio “aptitude”, que nos otorga acceso a los programas necesarios para la generación del certificado.

```
nairda@SRVPROFTPD: ~
nairda@SRVPROFTPD:~$ sudo apt-get install aptitude
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  aptitude-common libboost-iostreams1.74.0 libcwidget4 libsigc++-2.0-0v5 libxapian30
Paquetes sugeridos:
  apt-xapian-index aptitude-doc-en | aptitude-doc debtags tasksel libcwidget-dev xapian-tools
Se instalarán los siguientes paquetes NUEVOS:
  aptitude aptitude-common libboost-iostreams1.74.0 libcwidget4 libsigc++-2.0-0v5 libxapian30
0 actualizados, 6 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
Se necesita descargar 4.083 kB de archivos.
Se utilizarán 19,5 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [Y/n] y
Des:1 http://es.archive.ubuntu.com/ubuntu jammy/universe amd64 aptitude-common all 0.8.13-3ubuntu1 [1.719 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu jammy/main amd64 libboost-iostreams1.74.0 amd64 1.74.0-14ubuntu3 [245 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu jammy/main amd64 libsigc++-2.0-0v5 amd64 2.10.4-2ubuntu3 [12,1 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu jammy/universe amd64 libcwidget4 amd64 0.5.18-5build1 [306 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu jammy/universe amd64 libxapian30 amd64 1.4.18-4 [701 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu jammy/universe amd64 aptitude amd64 0.8.13-3ubuntu1 [1.100 kB]
Descargados 4.083 kB en 0s (9.199 kB/s)
Seleccionando el paquete aptitude-common previamente no seleccionado.
(Leyendo la base de datos ... 131982 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../0-aptitude-common-0.8.13-3ubuntu1_all.deb ...
Desempaquetando aptitude-common (0.8.13-3ubuntu1) ...
Seleccionando el paquete libboost-iostreams1.74.0:amd64 previamente no seleccionado.
Preparando para desempaquetar .../1-libboost-iostreams1.74.0-1.74.0-14ubuntu3_amd64.deb ...
Desempaquetando libboost-iostreams1.74.0:amd64 (1.74.0-14ubuntu3) ...
Seleccionando el paquete libsigc++-2.0-0v5:amd64 previamente no seleccionado.
Preparando para desempaquetar .../2-libsigc++-2.0-0v5-2.10.4-2ubuntu3_amd64.deb ...
Desempaquetando libsigc++-2.0-0v5:amd64 (2.10.4-2ubuntu3) ...
```

Tras intentar instalar los repositorios, comprobamos que ya estaban instalados en el sistema en la versión requerida.

```
nairda@SRVPROFTPD: ~
nairda@SRVPROFTPD:~$ sudo aptitude install openssl ca-certificates
openssl ya está instalado en la versión solicitada (3.0.2-0ubuntu1.7)
ca-certificates ya está instalado en la versión solicitada (20211016)
openssl ya está instalado en la versión solicitada (3.0.2-0ubuntu1.7)
ca-certificates ya está instalado en la versión solicitada (20211016)
No se instalará, actualizará o eliminará ningún paquete.
0 paquetes actualizados, 0 nuevos instalados, 0 para eliminar y 6 sin actualizar.
Necesito descargar 0 B de ficheros. Después de desempaquetar se usarán 0 B.

nairda@SRVPROFTPD:~$
```

### 2.2. Configuración del certificado

Antes de generar el certificado autofirmado, tenemos que generar el certificado público con la clave. Esto se hace mediante el comando “*openssl genrsa -aes128 -out private.key 2048*”, este comando indica que se va a generar una nueva clave pública, con nombre private key, con un algoritmo de encriptación de tipo AES128. Se podrían haber elegido otro tipo de encriptaciones como podrían haber sido SHA1 o en su defecto SHA256.

```
nairda@SRVPROFTPD: ~
nairda@SRVPROFTPD:~$ sudo openssl genrsa -aes128 -out private.key 2048
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
nairda@SRVPROFTPD:~$ ls -la | grep *.key
-rw----- 1 root root 1874 nov 20 19:07 private.key
nairda@SRVPROFTPD:~$
```

Tras generar la clave pública, se necesita hacer el “request” de la clave privada para que sea generada. Para ello se utiliza el comando “*openssl req -new -days 365 -key <NOMBRE DE LA KEY> -out request.csr*”. En este comando observamos como se le indica el periodo válido de la clave privada, es decir, cada cuanto tiempo va a ser válida y cuando hemos de renovarla

```
nairda@SRVPROFTPD: ~
nairda@SRVPROFTPD:~$ sudo openssl req -new -days 365 -key private.key -out request.csr
Ignoring -days without -x509; not generating a certificate
Enter pass phrase for private.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]:ES
State or Province Name (full name) [Some-State]:Spain
Locality Name (eg, city) []:Madrid
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Nairda
Organizational Unit Name (eg, section) []:section
Common Name (e.g. server FQDN or YOUR name) []:SRVPOFTP
Email Address []:adrian.alba@outlook.es

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:celebi
An optional company name []:NairdaLTD
nairda@SRVPROFTPD:~$
```

Cuando lanzamos el comando, nos va solicitando una serie de datos para introducirlos en el certificado, como pueden ser el nombre de la organización, donde se encuentra el certificado, y demás.

Una vez generada la clave privada, se procede a generar el certificado que vamos a utilizar en nuestro servidor FTP. Para ello, utilizando el comando “*openssl x509 -in request.csr -out certificate.crt -req -signkey private.key -days 365*”.

```
nairda@SRVPROFTPD:~$ sudo openssl x509 -in request.csr -out certificate.crt -req -signkey private.key -days 365
Enter pass phrase for private.key:
Certificate request self-signature ok
subject=C = ES, ST = Spain, L = Madrid, O = Nairda, OU = section, CN = SRVPOFTP, emailAddress = adrian.alba@outlook.es
nairda@SRVPROFTPD:~$
```

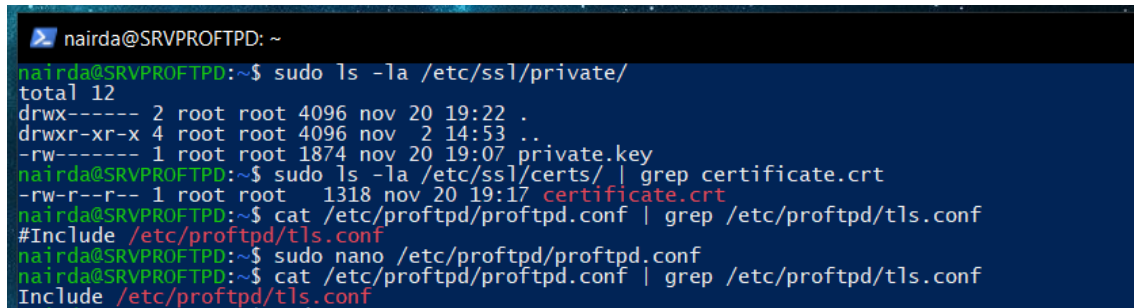
Una vez generado el certificado, movemos los ficheros que hemos generado a su correspondiente carpeta, la clave privada a la carpeta **/etc/ssl/private** y el certificado a **/etc/ssl/certs**.

```
nairda@SRVPROFTPD: ~
nairda@SRVPROFTPD:~$ sudo ls -la /etc/ssl/private/
total 12
drwx----- 2 root root 4096 nov 20 19:22 .
drwxr-xr-x 4 root root 4096 nov 2 14:53 ..
-rw----- 1 root root 1874 nov 20 19:07 private.key
nairda@SRVPROFTPD:~$ sudo ls -la /etc/ssl/certs/ | grep certificate.crt
-rw-r--r-- 1 root root 1318 nov 20 19:17 certificate.crt
nairda@SRVPROFTPD:~$
```

## 3. CONFIGURACIÓN DEL SERVIDOR FTP

### 3.1. Habilitar el módulo TLS

Tras la generación del certificado, hemos de habilitar el módulo que nos permita conectarnos a través de TLS al servidor FTP. Para ello, hemos de descomentar la línea que incluye el módulo TLS.

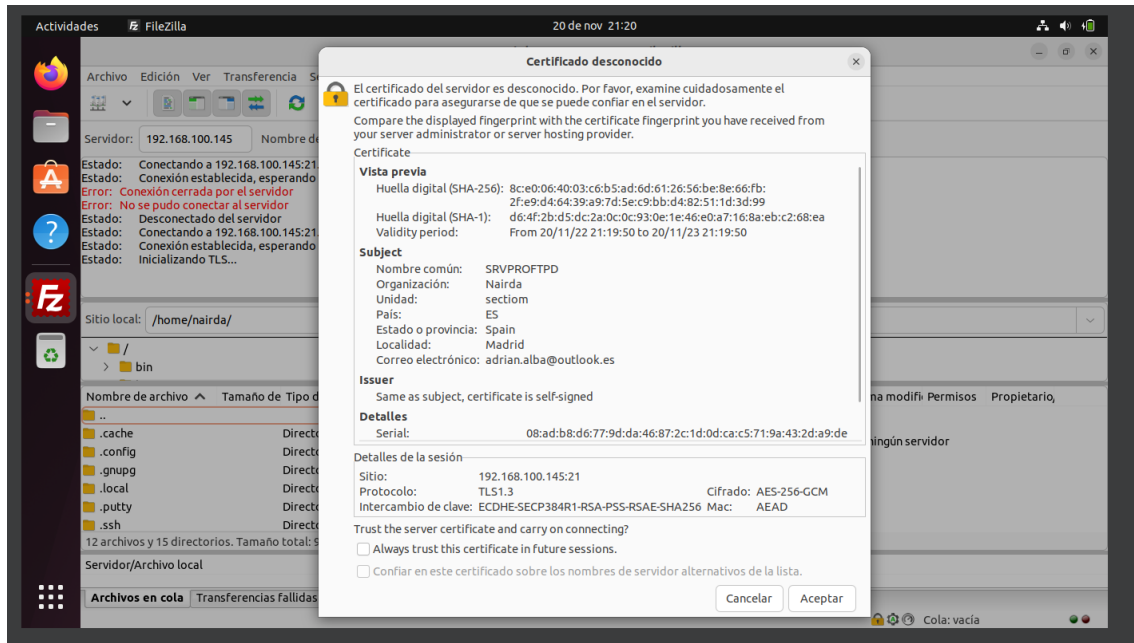
A terminal window with a dark blue background and white text. The prompt is 'nairda@SRVPROFTPD: ~'. The user runs 'sudo ls -la /etc/ssl/private/' showing files like 'private.key'. Then they run 'sudo ls -la /etc/ssl/certs/ | grep certificate.crt' showing 'certificate.crt'. Next, they run 'cat /etc/proftpd/proftpd.conf | grep /etc/proftpd/tls.conf' showing a commented line '#Include /etc/proftpd/tls.conf'. Finally, they run 'sudo nano /etc/proftpd/proftpd.conf' and 'cat /etc/proftpd/proftpd.conf | grep /etc/proftpd/tls.conf' showing the line 'Include /etc/proftpd/tls.conf'.

Se instala el módulo `proftpd-mod-crypto`, y se habilita el `tls` en el fichero de configuración **`/etc/proftpd/proftpd.conf`** y en **`/etc/proftpd/tls.conf`** se le indica donde encontrar los ficheros de la clave privada y del certificado, además de habilitar la comunicación TLS.

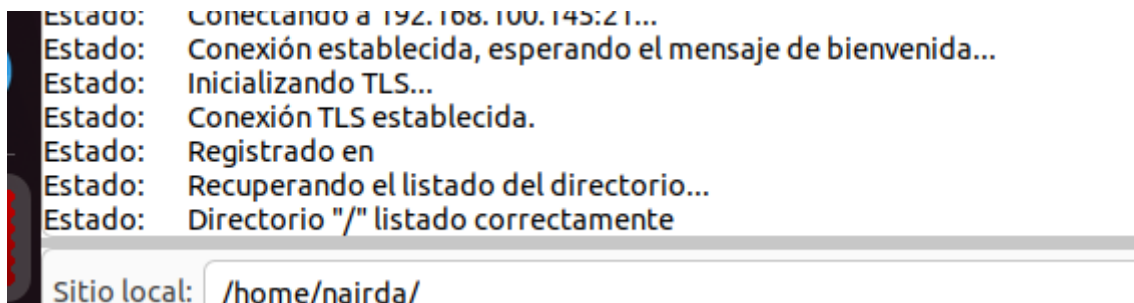
## 4. PRUEBAS REALIZADAS

### 4.1. Conexión

Cuando vamos a iniciar sesión en el servidor utilizando TLS por primera vez, nos pedirá añadir a nuestra lista de claves públicas la del servidor. Este paso es normal debido a que no lo teníamos previamente listado en nuestro fichero de claves públicas.



Tras guardar la clave, reinicia la conexión, y nos permite conectarnos al servidor utilizando un protocolo seguro de comunicación.



## 5. BIBLIOGRAFÍA

---

- <https://www.arubacloud.com/tutorial/how-to-create-a-self-signed-ssl-certificate-on-ubuntu-18-04.aspx>
- [https://www.server-world.info/en/note?os=Ubuntu\\_22.04&p=ssl&f=1](https://www.server-world.info/en/note?os=Ubuntu_22.04&p=ssl&f=1)