

PRÁCTICA 3

Seguridad (Certificados SSL)



UNIVERSIDAD DE GRANADA

Titulación: Ingeniería Informática + ADE

FLORIN EMANUEL TODOR GLIGA

ÍNDICE

1. [Tareas Básicas](#)
 - a. [B1. Preparación del Entorno de Trabajo:](#)
 - b. [B2. Creación de Certificados SSL](#)
 - c. [B3. Configuración de Servidores Web Apache y Nginx con SSL](#)
 - d. [B4. Configuración del Balanceador de Carga Nginx con SSL](#)
 - e. [B5. Docker Compose para la Granja Web con SSL](#)
 - f. [B6. Verificación y Pruebas del Escenario con SSL](#)
2. [Tareas Avanzadas](#)
 - a. [A1. Exploraciones Avanzadas de creación de certificados SSL](#)
 - b. [A2. Optimización de la configuración SSL en los servidores web](#)
 - c. [A3. Configuración de Caché y Tickets de Sesión SSL en el balanceador](#)
 - d. [A4. Optimización de conexiones HTTPS y cifrado en el balanceador](#)
3. [Uso de Inteligencia Artificial Generativa](#)

Tareas básicas

De forma previa a la práctica, debo de comentar que estoy usando en todo momento el docker compose del balanceador de nginx (docker-compose_nginx_balanceador.yaml), aunque voy a adaptarlo a los demás compose (en otro momento ya que no son nginx los balanceadores). Por ello en todas las pruebas de las capturas estoy utilizando **./init.sh -u nginx, ejecutando el balanceador de carga nginx con la estrategia de round-robin.**

B1. Preparación del Entorno de Trabajo

Antes de comenzar con la práctica. Quiero comentar que yo estoy implementando de forma consecutiva cada práctica, es decir, en la práctica 2 utilizo la implementación de la práctica 1 y en este caso en la práctica 3 usaré toda la implementación de la práctica 2, sin embargo, implementaré a su vez lo que se nos solicita en esta práctica. Esto lo hago para tener en sí una evolución continua de todo el proyecto con las mismas implementaciones anteriores (me parece interesante a nivel personal).

Además, comentar, que en cada práctica, aunque utilice el código de las implementaciones de las anteriores prácticas, se basan en nuevas imágenes de docker.

Por lo tanto, permito conectarme por http y https con esta práctica.

Para esta parte muestro el tree (de los directorios creados) de la organización actual para los ejercicios básicos de esta práctica:

```
● > tree -d
.
├── file_sd
├── logs_apache
├── logs_envoy
├── logs_escalado
├── logs_haproxy
├── logs_nginx
├── logs_traefik
├── P3-flotodor-apache
│   └── apache_config
├── P3-flotodor-certificados
├── P3-flotodor-envoy
├── P3-flotodor-haproxy
│   └── config_balanceador
├── P3-flotodor-nginx
│   ├── config_balanceador
│   └── config_webs
├── P3-flotodor-traefik
└── web_flotodor
```

Vemos la creación del directorio de certificados, la de apache (que es la que llevo usando todo el rato) y la de nginx (aunque en esta hago la diferenciación entre el config de mis webs nginx y del balanceador).

B2. Creación de Certificados SSL

[illegible]

Como podemos ver, creamos el certificado SSL con las especificaciones que se nos solicita. Respecto a los parámetros:

- X509 lo utilizamos para crear un certificado autofirmado
- nodes lo utilizamos para dejar la clave sin passphrase, es decir, que no nos pida la clave.
- newkey rsa:2048 para generar una clave RSA de 2048 bits
- sha256 utilizamos este cifrado para la firma
- days 365 validez de 365 del certificado
- keyout para indicar cómo queremos la salida de la clave
- out para indicar cómo queremos la salida del certificado
- en subj nos encontramos con las últimas especificaciones que se nos solicita en la práctica, localidad, nombre, correo, etc.

Comprobamos que se ha creado correctamente el certificado y su clave pública:

```

●) openssl x509 -in certificado_flotodor.crt -text -noout
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      21:bc:3d:da:79:9a:6c:19:78:63:cd:5c:53:fd:3d:63:20:8c:75:2e
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: C = ES, ST = Granada, L = Granada, O = SWAP, OU = Practica 3, CN = "Florin Emanuel Todor Gliga", emailAddress = flotodor@correo.ugr.es
    Validity
      Not Before: Apr 24 21:48:52 2025 GMT
      Not After : Apr 24 21:48:52 2026 GMT
    Subject: C = ES, ST = Granada, L = Granada, O = SWAP, OU = Practica 3, CN = "Florin Emanuel Todor Gliga", emailAddress = flotodor@correo.ugr.es
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
      Public-Key: (2048 bit)
      Modulus:
        00:c0:3c:c4:d9:b6:6d:db:5f:e3:43:3b:5e:e9:77:
        aa:3d:4b:3e:e8:81:15:08:03:08:f9:f5:87:9e:9b:
        94:cd:a2:08:1a:90:52:0c:24:61:24:5f:8c:02:f3:
        01:e7:47:eb:3d:dd:69:b8:3b:5c:68:18:61:05:80:
        34:06:ee:de:a4:92:b7:c4:4b:73:87:26:40:96:74:
        e0:18:8e:0c:de:92:a9:4f:d0:0e:87:03:f7:cd:32:
        5e:95:96:a0:49:a5:d9:1a:53:6b:00:5d:00:df:8e:
        1a:f7:24:0c:08:c5:46:06:30:63:56:29:01:72:d7:
        18:dd:52:75:5b:9c:2a:4a:b4:c2:20:f6:fc:7e:2d:
        88:ae:2e:ad:36:ad:0d:00:01:e2:7d:99:ab:17:f8:
        7b:9e:62:ec:1c:45:ab:e5:8b:7e:ee:d7:8a:e0:3f:
        5e:f3:ec:0e:63:00:4b:e9:07:6d:9e:5f:43:32:79:
        59:2a:d4:21:0f:0d:85:38:43:2d:5f:2d:8b:6e:35:
        cb:05:1d:41:f2:5c:ef:66:1f:2b:32:c6:b0:1d:6e:
        51:0f:93:27:c0:2b:51:ad:9e:56:e8:57:ff:c0:76:
        59:6a:9b:52:56:2b:9b:f8:8e:b8:fb:a6:1e:f0:34:
        cc:5e:99:26:c0:a2:a8:7f:37:78:c3:f8:05:b0:bc:
        86:f9
      Exponent: 65537 (0x10001)

```

Como podemos observar, está todo en orden.

Por último comprobamos que la clave esté bien creada:

```

$ openssl rsa -in certificado_flotador.key -check -noout
Could not open file or uri for loading private key from certificado_flotador.key
4077B3B2267C0000:error:16000069:STORE routines:ossl_store_get0_loader_int:unregistered scheme:../crypto/store/store_register.c:237:scheme=file
4077B3B2267C0000:error:80000000:system library:BIO_new_file:Permission denied:../crypto/bio/bss_file.c:67:calling fopen(certificado_flotador.key, rb)
4077B3B2267C0000:error:10000002:BIO routines:BIO_new_file:system lib:../crypto/bio/bss_file.c:77:
$ sudo openssl rsa -in certificado_flotador.key -check -noout
RSA key ok

```

Como podemos ver, al principio no nos deja comprobarlo por el cambio de permisos.

B3. Configuración de Servidores Web Apache y Nginx con SSL

Aunque en el ejercicio se nos pide solo para los servidores apache, como en mi implementación tengo apache y nginx, voy a implementarlo para ambos.

APACHE:

Primero creamos el fichero de flotador-apache-ssl.conf:

```

P3-flotador-apache > $ cat flotador-apache-ssl.conf
1  <IfModule mod_ssl.c>
2  <VirtualHost *:443>
3      DocumentRoot /var/www/html
4
5      SSLEngine on
6      SSLCertificateFile      /etc/apache2/ssl/certificado_flotador.crt
7      SSLCertificateKeyFile    /etc/apache2/ssl/certificado_flotador.key
8
9      <Directory /var/www/html>
10         Options Indexes FollowSymLinks # Permite la navegación por directorios
11         AllowOverride All # Permite el uso de .htaccess
12         Require all granted # Permite el acceso a todos los usuarios, incluidos los no autenticados
13     </Directory>
14
15     ErrorLog  ${APACHE_LOG_DIR}/ssl_error.log
16     CustomLog ${APACHE_LOG_DIR}/ssl_access.log combined
17 </VirtualHost>
18 </IfModule>
19

```

Como vemos, es la información que se nos facilita en el guión más la parte añadida de los logs y las modificaciones sobre cómo se debe de comportar apache con las peticiones, es decir, permitir la navegación por directorios en el caso de que no exista algún index. Permite archivos .htaccess y permite el acceso a todos los usuarios.

Los comentarios de esas tres líneas de código las he borrado para evitar errores.

Por otra parte, el código que inserto en el dockerfile de apache es:

```
# -----
# Parte SSL: certificados + configuración SSL
# -----

RUN mkdir -p /etc/apache2/ssl/
COPY P3-flotodor-certificados/certificado_flotodor.crt /etc/apache2/ssl/certificado_flotodor.crt
COPY P3-flotodor-certificados/certificado_flotodor.key /etc/apache2/ssl/certificado_flotodor.key

RUN chmod 600 /etc/apache2/ssl/certificado_flotodor.crt
COPY P3-flotodor-apache/flotodor-apache-ssl.conf /etc/apache2/sites-available/flotodor-apache-ssl.conf
RUN a2enmod ssl && a2dissite default-ssl && a2ensite flotodor-apache-ssl

EXPOSE 80 443 9100
ENTRYPOINT ["/usr/local/bin/entrypoint_apache.sh"]
```

Lo primero que hago es enviar los ficheros que vamos a usar al directorio tmp y lo posterior es lo que se nos indica en el guión, es decir, colocar el certificado y clave en el directorio de apache, cambiar los permisos de dichos ficheros, deshabilitar las configuraciones por defecto de los vhost de apache y activar la que hemos creado (todo esto para que se reconozca el certificado creado).

Servidores nginx:

Por lo que he visto, y preguntado a gpt, tengo que copiar los certificados en /etc/nginx/ssl y configurar el default de nginx para incorporar el https:

```
9 && rm -rf node_exporter+
10
11 # Exponemos el puerto 80 para HTTP y 443 para HTTPS y 9100 para el node exporter
12 EXPOSE 80 443 9100
13
14 RUN mkdir -p /etc/nginx/ssl
15
16 # Copiamos el certificado y la clave privada
17 COPY ./P3-flotodor-certificados/certificado_flotodor.crt /etc/nginx/ssl/
18 COPY ./P3-flotodor-certificados/certificado_flotodor.key /etc/nginx/ssl/
19
20 # Reemplazamos por completo el ENTRYPOINT original de nginx
21 CMD ["/entrypoint.sh"]
```

En el default añado esta parte:

```
26 server {
27     listen 443 ssl default_server;
28     listen [::]:443 ssl default_server;
29
30     root /usr/share/nginx/html;
31     index index.php index.html;
32
33     server_name _;
34
35     ssl_certificate /etc/nginx/ssl/certificado_flotodor.crt;
36     ssl_certificate_key /etc/nginx/ssl/certificado_flotodor.key;
37
38     location / {
39         try_files $uri $uri/ /index.php?$query_string;
40     }
41
42     location ~ \.php$ {
43         include snippets/fastcgi-php.conf;
44         fastcgi_pass unix:/run/php/php8.3-fpm.sock;
45         fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
46         include fastcgi_params;
47     }
48
49     location ~ /\. {
50         deny all;
51     }
52 }
```

B4. Configuración del Balanceador de Carga Nginx con SSL

Tengo que comentar que crearé el fichero de nginx-ssl.conf pero solamente para el balanceador además de modificar el dockerfile del balanceador. Esto lo comento debido a que en la carpeta de las configuraciones de nginx tengo tanto la configuración de los servidores webs nginx como la configuración del balanceador de nginx.

Por lo que primero voy a crear el fichero de flotodor-nginx-ssl.conf:

```

1  server {
2      listen 443 ssl;
3
4      # Ruta a los certificados dentro del contenedor
5      ssl_certificate      /etc/nginx/ssl/certificado_flotodor.crt;
6      ssl_certificate_key  /etc/nginx/ssl/certificado_flotodor.key;
7
8      location / {
9          proxy_pass http://backend_flotodor;
10         proxy_set_header Cookie $http_cookie;
11         proxy_hide_header Set-Cookie;
12     }
13 }

```

Dockerfile del balanceador:

```

1  FROM nginx:latest
2
3  # -----
4  # Paquetes básicos
5  RUN apt-get update && apt-get upgrade -y
6  RUN apt install -y net-tools iputils-ping iptables
7  # -----
8
9  # Configuración de Nginx: limpiar index.html y nginx.conf
10 RUN rm -f /usr/share/nginx/html/index.html
11 RUN rm -f /etc/nginx/nginx.conf
12
13 # Creamos directorios para SSL (vacíos)
14 RUN mkdir -p /etc/nginx/ssl
15
16 # -----
17 # Exponemos puertos para HTTP y HTTPS
18 EXPOSE 80 443
19
20 # Lanzamos Nginx en primer plano
21 CMD ["nginx", "-g", "daemon off;"]
22

```


B5. Docker Compose para la Granja Web con SSL

Docker compose del balanceador de carga de nginx:

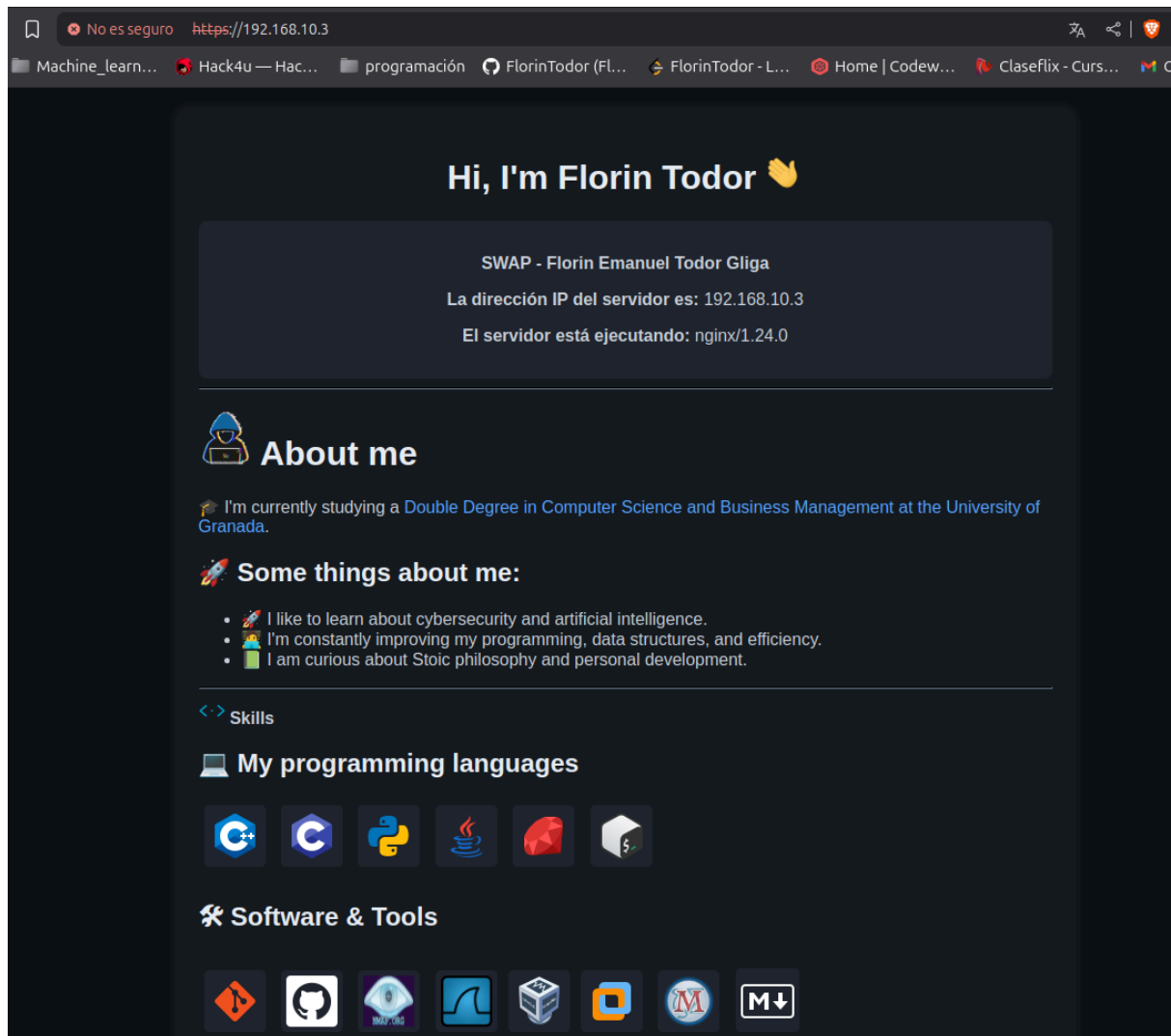
```

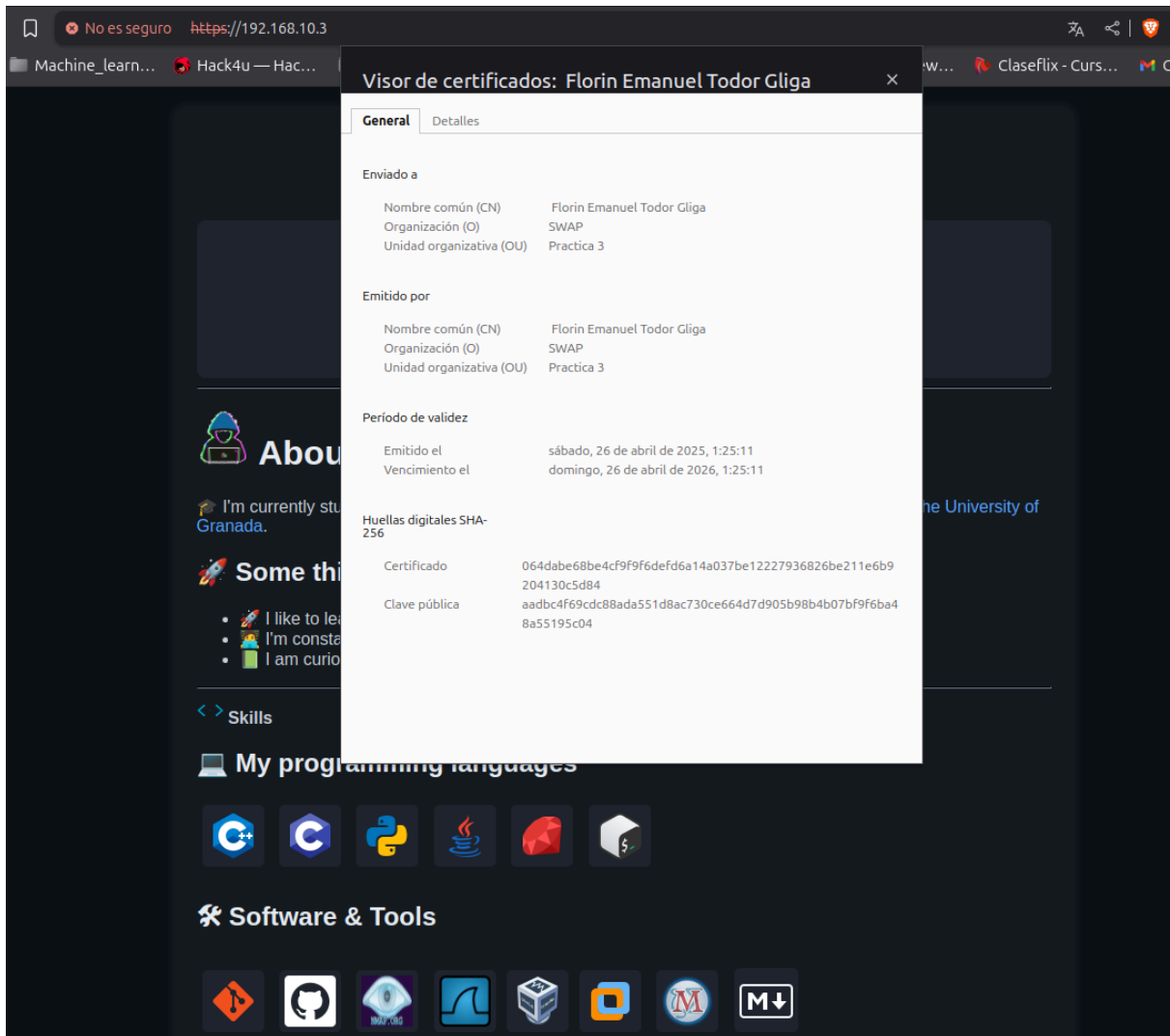
1  # Datos comunes para todos los servicios de apache
2  x-common-apache-config: &common-apache-config
3  image: flotodor-apache-image:p3
4  restart: always
5  volumes:
6    - ./web_flotodor:/var/www/html
7    - ./logs_apache:/var/log/apache2
8
9
10 # Datos comunes para todos los servicios de nginx
11 x-common-nginx_web-config: &common-nginx_web-config
12 image: flotodor-nginx_web-image:p3
13 restart: always
14 volumes:
15   - ./web_flotodor:/usr/share/nginx/html:ro
16   - ./logs_nginx:/var/log/nginx
17
18
19 x-common-nginx_balanceador-config: &common-nginx_balanceador-config
20 image: flotodor-nginx_balanceador-image:p3
21 restart: always
22 volumes:
23   - ./P3-flotodor-nginx/config_balanceador/flotodor-nginx-ssl.conf:/etc/nginx/nginx.conf # << Montar nginx.conf SSL
24   - ./P3-flotodor-certificados:/etc/nginx/ssl # << Montar certificados SSL
25   - ./logs_nginx:/var/log/nginx
26
27
28 - "8088:80"
29
30
31 Run Service
32 nginx_balanceador:
33   <<: *common-nginx_balanceador-config
34   container_name: nginx_balanceador
35   environment:
36     - SERVER_NAME=nginx_balanceador
37   networks:
38     red_web:
39       ipv4_address: 192.168.10.50
40   ports:
41     - "80:80"
42     - "443:443"
43   depends_on:
44     - web1
45     - web2
46     - web3
47     - web4
48     - web5
49     - web6
50     - web7
51     - web8
52
53
54 networks:
55   red_web:

```

B6. Verificación y Pruebas del Escenario con SSL

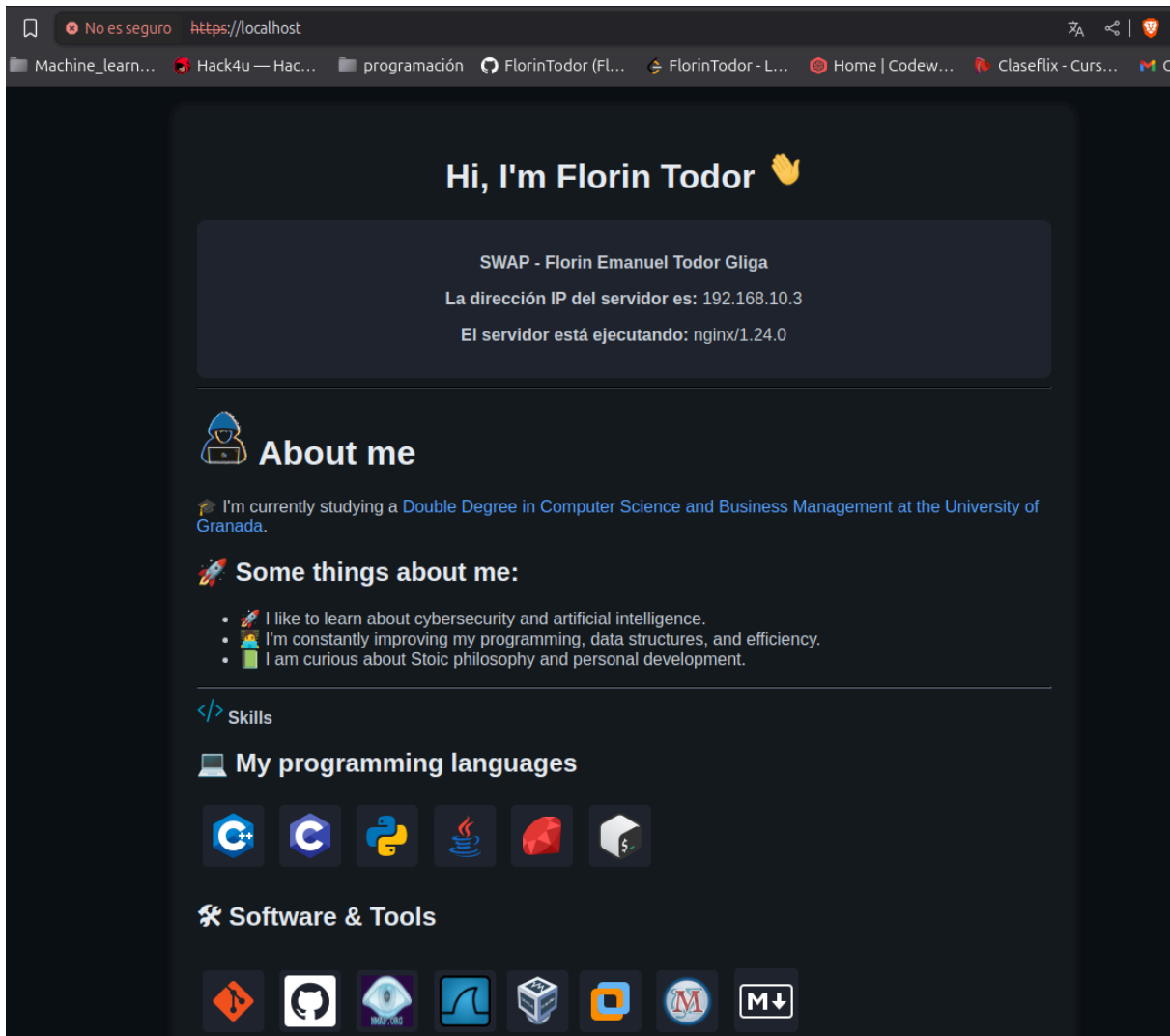
Primero voy a realizar pruebas a los propios servidores web para comprobar que funcionan los certificados:

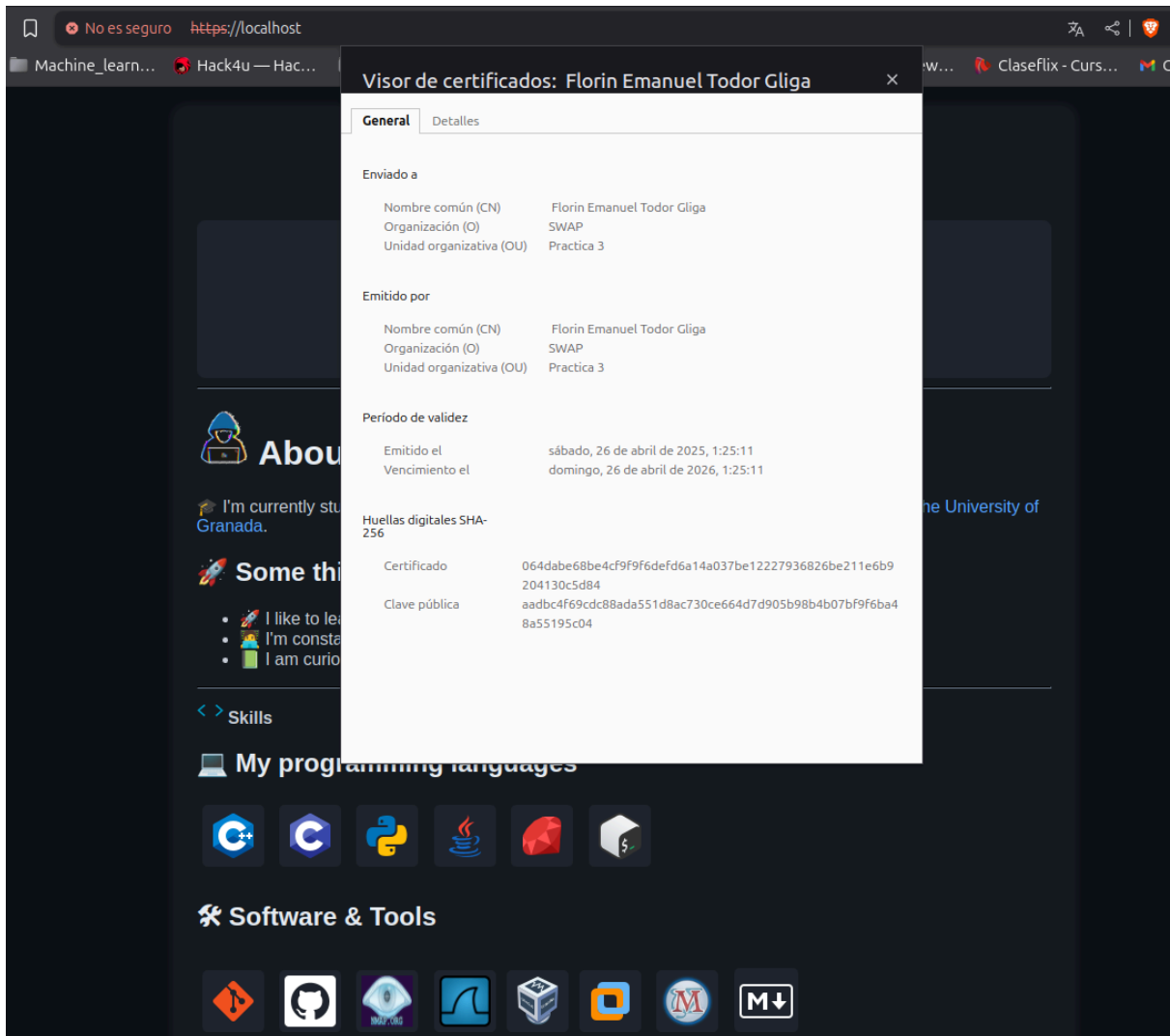


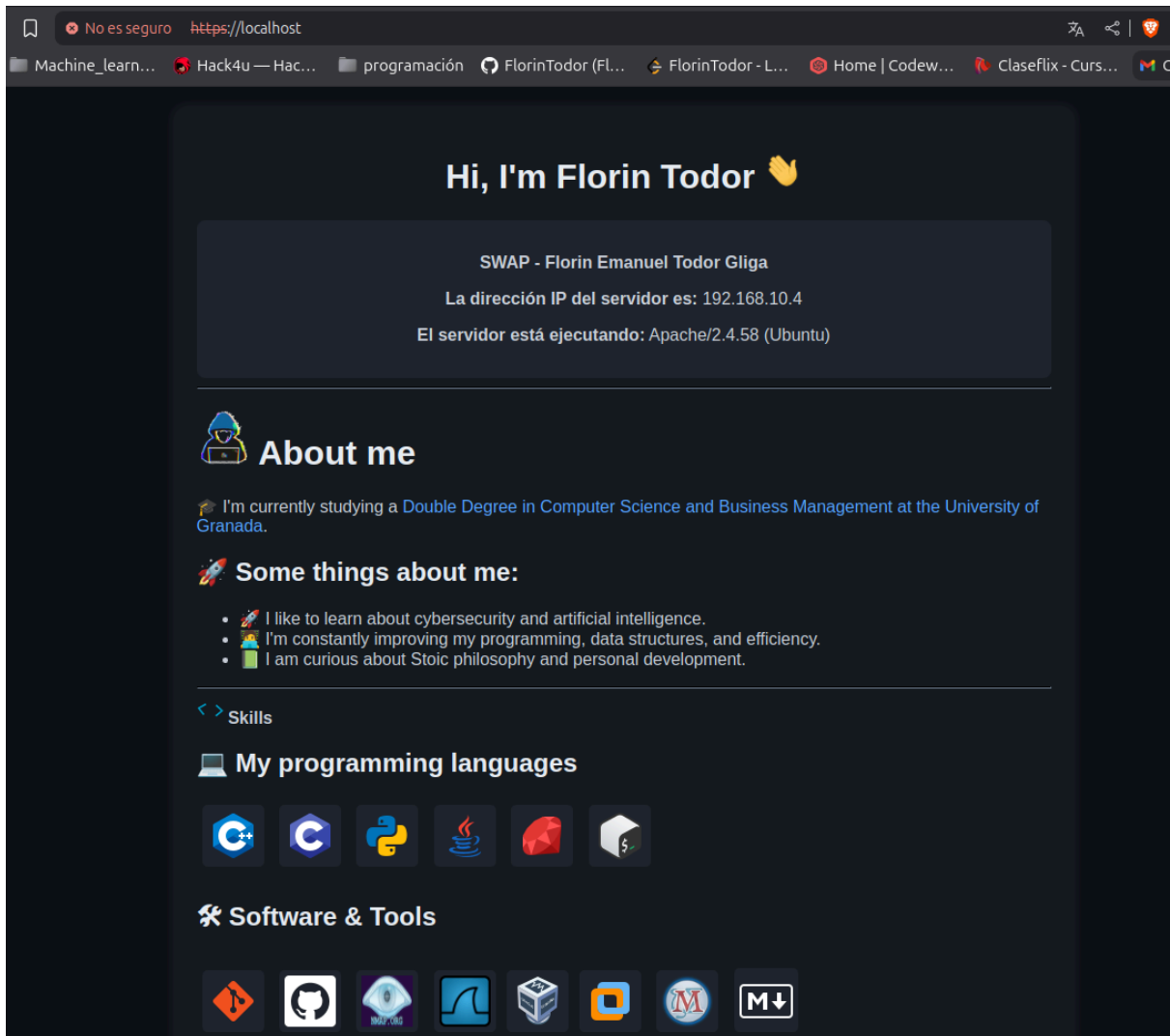


Vemos que funciona, igual que al probarlo en <https://192.168.10.2> (servidor apache), también funciona.

Por lo tanto, voy a comprobar ahora que el balanceador, realiza el balanceo de carga a través de https y que interpreta correctamente el certificado ssl.







Vemos que realiza de forma correcta el balanceo y que reconoce a su vez el certificado ssl creado.

TAREAS AVANZADAS

A1. Exploraciones Avanzadas de creación de certificados SSL

Como no entendía el funcionamiento de esta actividad lo he hablado con GPT y me ha parecido interesante, procedo a la implementación:

Para esta parte, voy a crear un certificado y clave raíz, posteriormente creo un certificado intermedio (subca) firmando la subca con la CA raíz y por último creamos el certificado que usaremos en los servidores webs, el cual está firmado por el subca.

Se nos quedará la siguiente cadena de confianza:

Servidor web -> SubCA -> CA raíz.

Todos los certificados y claves los voy a incluir en el directorio p3-flotodor-certificados.

Creamos primero el certificado y clave de CA raíz:

```

• > cd CA
• > ls
• > openssl genrsa -out ca-raiz.key 4096
• > openssl req -x509 -new -nodes -key ca-raiz.key -sha256 -days 3650 -out ca-raiz.crt
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]:Granada
Locality Name (eg, city) []:Granada
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Universidad de Granada
Organizational Unit Name (eg, section) []:Universidad de Granada
Common Name (e.g. server FQDN or YOUR name) []:Florin Emanuel Todor Gliga
Email Address []:flotodor@correo.ugr.es
• > ls
  ca-raiz.crt  ca-raiz.key
  
```

Ahora, creamos el la clave privada del SubCA, la solicitud de firma de certificado (CSR) para la SubCA y la firmamos con la CA raíz.

```

> cd ../SubCA
> ls
> openssl genrsa -out subca.key 4096
> openssl req -new -key subca.key -out subca.csr
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]:Granada
Locality Name (eg, city) []:Granada
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Universidad de Granada
Organizational Unit Name (eg, section) []:Universidad de Granada
Common Name (e.g. server FQDN or YOUR name) []:Florin Emanuel Todor Gliga
Email Address []:flotodor@correo.ugr.es

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:SWAP1234
An optional company name []:
> openssl x509 -req -in subca.csr -CA ca-raiz.crt -CAkey ca-raiz.key -CAcreateserial -out subca.crt -days 1825 -sha256

Certificate request self-signature ok
subject=C = ES, ST = Granada, L = Granada, O = Universidad de Granada, OU = Universidad de Granada, CN = Florin Emanuel Todor Gliga, emailAddress = flotodor@correo.ugr.es
Could not open file or uri for loading CA certificate from ca-raiz.crt
4077486719710000:error:16000069:STORE routines:ossl_store_get0_loader_int:unregistered scheme:../crypto/store/store_register.c:237:scheme=file
4077486719710000:error:80000002:system library:file_open:No such file or directory:../providers/implementations/storemgmt/file_store.c:267:calling stat(ca-raiz.crt)
Unable to load CA certificate
> openssl x509 -req -in subca.csr -CA ../CA/ca-raiz.crt -CAkey ../CA/ca-raiz.key -CAcreateserial -out subca.crt -days 1825 -sha256

Certificate request self-signature ok
subject=C = ES, ST = Granada, L = Granada, O = Universidad de Granada, OU = Universidad de Granada, CN = Florin Emanuel Todor Gliga, emailAddress = flotodor@correo.ugr.es

```

Por último, nos quedaría crear la clave para los servidores y el csr, además de firmarlo con la SubCA.

```

> ls
> openssl genrsa -out servidor-web.key 2048
> openssl req -new -key servidor-web.key -out servidor-web.csr
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]:Granada
Locality Name (eg, city) []:Granada
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Universidad de Granada
Organizational Unit Name (eg, section) []:Universidad de Granada
Common Name (e.g. server FQDN or YOUR name) []:Florin Emanuel Todor Gliga
Email Address []:flotodor@correo.ugr.es

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:SWAP1234
An optional company name []:
> openssl x509 -req -in servidor-web.csr -CA subca.crt -CAkey subca.key -CAcreateserial -out servidor-web.crt -days 825 -sha256

Certificate request self-signature ok
subject=C = ES, ST = Granada, L = Granada, O = Universidad de Granada, OU = Universidad de Granada, CN = Florin Emanuel Todor Gliga, emailAddress = flotodor@correo.ugr.es
Could not open file or uri for loading CA certificate from subca.crt
40C70E4316750000:error:16000069:STORE routines:ossl_store_get0_loader_int:unregistered scheme:../crypto/store/store_register.c:237:scheme=file
40C70E4316750000:error:80000002:system library:file_open:No such file or directory:../providers/implementations/storemgmt/file_store.c:267:calling stat(subca.crt)
Unable to load CA certificate
> openssl x509 -req -in servidor-web.csr -CA ../SubCA/subca.crt -CAkey ../SubCA/subca.key -CAcreateserial -out servidor-web.crt -days 825 -sha256

Certificate request self-signature ok
subject=C = ES, ST = Granada, L = Granada, O = Universidad de Granada, OU = Universidad de Granada, CN = Florin Emanuel Todor Gliga, emailAddress = flotodor@correo.ugr.es

```


Entonces, nuestro directorio de trabajo de los certificados se nos quedaría de la siguiente manera:

```

• > tree P3-flotodor-certificados
P3-flotodor-certificados
├── CA
│   ├── ca-raiz.crt
│   ├── ca-raiz.key
│   └── ca-raiz.srl
├── certificado_flotodor.crt
├── certificado_flotodor.key
├── SubCA
│   ├── subca.crt
│   ├── subca.csr
│   ├── subca.key
│   └── subca.srl
└── Webs
    ├── servidor-web.crt
    ├── servidor-web.csr
    └── servidor-web.key

4 directories, 12 files

~/Escritorio/SWAP/P3 | on main !7

```

Teniendo en cuenta que para esta tarea avanzada usamos los elementos de los directorios CA, SubCA y Webs.

Además, debemos de concatenar la información de los certificados para poder mostrar la jerarquía de la cadena completa.

Para ello creamos en webs el certificado de **servidor-web-full.crt**:

```

> cat Webs/servidor-web.crt SubCA/subca.crt CA/ca-raiz.crt > Webs/servidor-web-full.crt

~/Escritorio/SWAP/P3/P3-flotodor-certificados | on main !18

```

Por último, nos quedaría modificar los docker compose y ficheros de configuración para que reconozca esta cadena.

Respecto a los servidores webs nginx:

En el fichero de default.

```

ssl_certificate /etc/nginx/ssl/Webs/servidor-web-full.crt;
ssl_certificate_key /etc/nginx/ssl/Webs/servidor-web.key;
ssl_trusted_certificate /etc/nginx/ssl/CA/ca-raiz.crt;

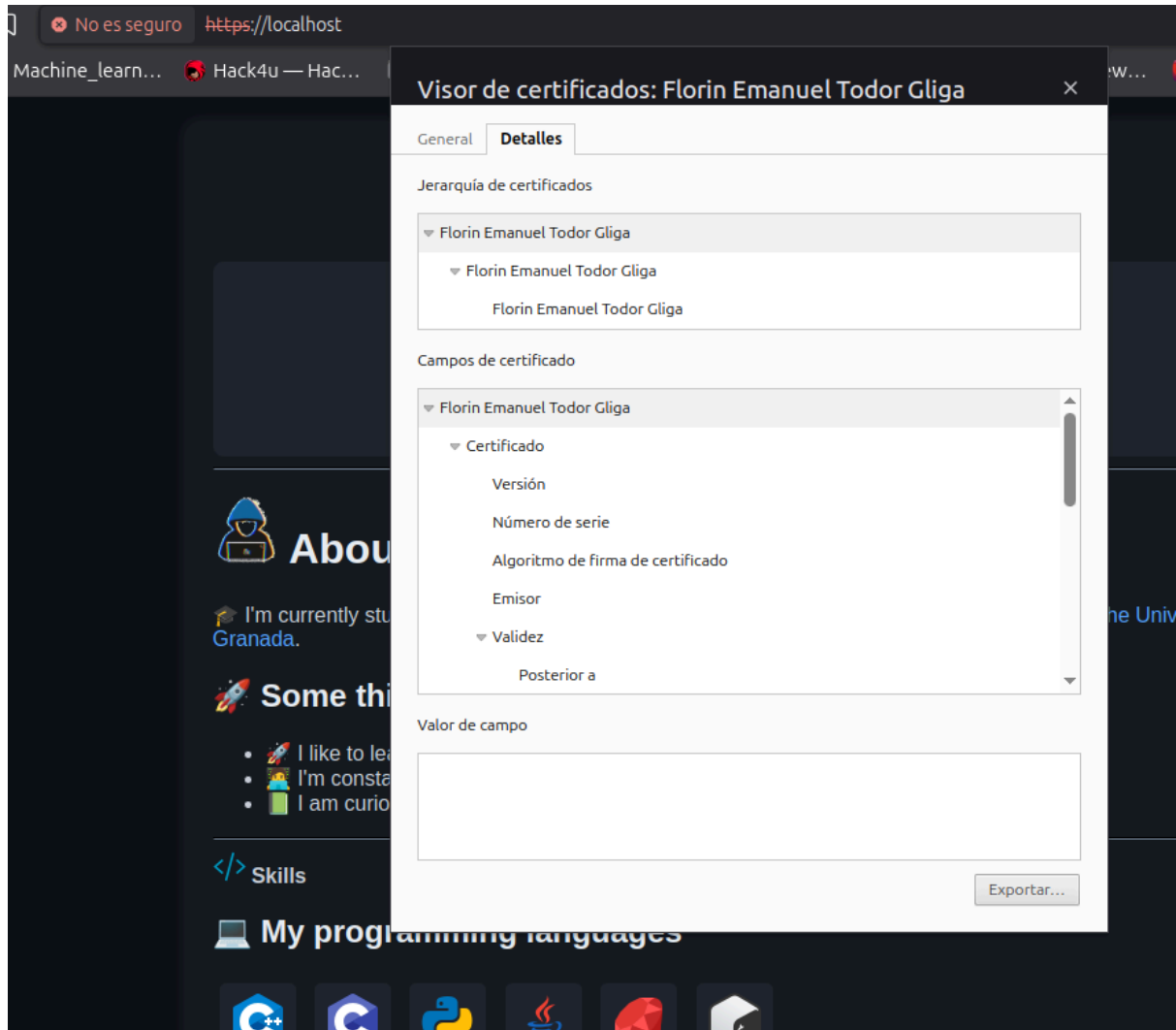
```

Respecto a los servidores webs apache:

```
SSLEngine on
SSLCertificateFile /etc/apache2/ssl/Webs/servidor-web-full.crt
SSLCertificateKeyFile /etc/apache2/ssl/Webs/servidor-web.key
SSLCertificateChainFile /etc/apache2/ssl/CA/ca-raiz.crt
```

Respecto al balanceador es lo mismo que el de nginx pero en el fichero de nginx.conf

Podemos comprobar desde localhost que se muestra la jerarquía de los certificados:



Realmente debería de haberle cambiado el nombre para indicar cuál es CA, SubCA y webs, pero supongo que se entiende que hay 3 certificados y el orden de cada uno en la cadena de autenticación.

A2: Optimización de la configuración SSL en los servidores web

Para esta parte voy a desactivar los protocolos inseguros como son TLSv1.0 y 1.1, debido a que son vulnerables a ataques como BEAST, POODLE

<https://mobiledatas.com/2023/09/05/el-riesgo-de-seguridad-al-utilizar-tls-en-el-correo-electronico/>

Además, voy a activar los protocolos TLSv1.2 y TLSv1.3 que son más recomendados.

Por otro lado, voy a forzar cifrados fuertes como el de AES-GCM, ya que los que se comentan en la práctica (RC4,3DES,MD5) tienen varias vulnerabilidades.

Por lo tanto, para la implementación tenemos que modificar los ficheros ssl.conf:

```
SSLProtocol -all +TLSv1.2 +TLSv1.3
SSLCipherSuite HIGH:!aNULL:!MD5:!3DES:!RC4:!DES:!EXP:!eNULL
SSLHonorCipherOrder on
```

Con esto indicamos que los protocolos a usar solo sean las de TLSv1.2 y 1.3, además de activar los cifrados fuertes (HIGH) y eliminando los débiles. Por otro lado, obligamos a que el servidor imponga su lista de cifrados seguros, es decir, que el usuario no puede elegir.

Eso es para Apache, para nginx tenemos lo siguiente (para los servidores webs), se encuentra en el fichero de default:

```
ssl_protocols TLSv1.2 TLSv1.3;
ssl_ciphers 'ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305';
ssl_prefer_server_ciphers on;
```

A3: Configuración de Caché y Tickets de Sesión SSL en el balanceador

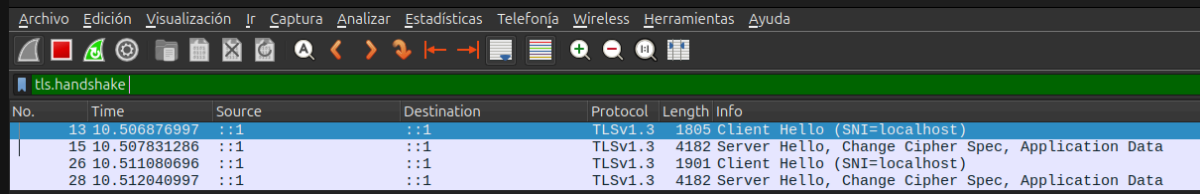
Sabía lo que era la caché pero los tickets de sesión no y la verdad es que es interesante (sobre todo la parte de los tickets). Paso a la parte de la implementación:

```
ssl_session_cache shared:SSL:10m;
ssl_session_timeout 20m;
ssl_session_tickets on;
```

Al añadir esto en el fichero de ssl.conf estamos indicando que:

Guarde sesiones SSL en una caché llamada SSL, con 10 MB de RAM reservados, además de que las sesiones de caché sean válidas durante 20 minutos y que Nginx use y entregue tickets de sesión SSL.

Voy a hacer una comprobación del funcionamiento con Wireshark:



No.	Time	Source	Destination	Protocol	Length	Info
13	10.506876997	:::1	:::1	TLSv1.3	1805	Client Hello (SNI=localhost)
15	10.507831286	:::1	:::1	TLSv1.3	4182	Server Hello, Change Cipher Spec, Application Data
26	10.511080696	:::1	:::1	TLSv1.3	1901	Client Hello (SNI=localhost)
28	10.512040997	:::1	:::1	TLSv1.3	4182	Server Hello, Change Cipher Spec, Application Data

Vemos que se está usando TLSv1.3.

GPT me comenta que en esta versión ya no existe la “caché de sesión tradicional” basada en Session ID como en TLSv1.2. Sino, que ahora solo existen los Session Tickets para reanudar las conexiones. Voy a comprobar si existe.

The image shows a Wireshark packet capture of a TLS handshake. The left pane displays a list of packets, with packet 1225 (Time: 358.351723600, Source: ::1, Destination: ::1) selected. The middle pane shows the details of this packet, which is a 'Handshake Protocol: Client Hello'. The right pane shows the raw bytes of the packet in hexadecimal and ASCII.

Packet List:

No.	Time	Source	Destination
1225	358.351723600	::1	::1
1227	358.352532087	::1	::1
1236	358.353363567	::1	::1
1238	358.354400525	::1	::1

Packet Details:

- Handshake Protocol: Client Hello (1)
 - Length: 2013
 - Version: TLS 1.2 (0x0303)
 - Random: 9f73a6eac4b7f1da662eb7b06c332b6e50b8dbf225b2447e877d48468c61a25b
 - Session ID Length: 32
 - Session ID: 67219023b53c184fe061415145461564a93821f09759a68ec6b5e6c837685775
 - Cipher Suites Length: 32
 - Cipher Suites (16 suites)
 - Compression Methods Length: 1
 - Compression Methods (1 method)
 - Extensions Length: 1908
 - Extension: Reserved (GREASE) (len=0)
 - Extension: key_share (len=1263) Unknown (4588), x25519
 - Extension: encrypted_client_hello (len=186)
 - Extension: signature_algorithms (len=18)
 - Extension: compress_certificate (len=3)
 - Extension: Unknown type 17613 (len=5)
 - Extension: extended_master_secret (len=0)
 - Extension: status_request (len=5)
 - Extension: renegotiation_info (len=1)
 - Extension: supported_versions (len=7) TLS 1.3, TLS 1.2
 - Extension: server_name (len=14) name=localhost
 - Extension: application_layer_protocol_negotiation (len=14)
 - Extension: signed_certificate_timestamp (len=0)
 - Extension: supported_groups (len=12)
 - Extension: ec_point_formats (len=2)
 - Extension: psk_key_exchange_modes (len=2)
 - Extension: session_ticket (len=0)
 - Extension: Reserved (GREASE) (len=1)
 - Extension: pre_shared_key (len=299)
 - [JA4: t13d1517h2_8daaf6152771_b6f405a00624]
 - [JA4_r: t13d1517h2_002f,0035,009c,009d,1301,1302,1303,c013,c014,c02b,c02c,c02f,c030,cca8,cca9]
 - [JA3 Fullstring: 771,4865-4866-4867-49195-49196-49200-52393-52392-49171-49172-156-157-158-159-160-161-162-163-164-165-166-167-168-169-170-171-172-173-174-175-176-177-178-179-180-181-182-183-184-185-186-187-188-189-190-191-192-193-194-195-196-197-198-199-200-201-202-203-204-205-206-207-208-209-210-211-212-213-214-215-216-217-218-219-220-221-222-223-224-225-226-227-228-229-230-231-232-233-234-235-236-237-238-239-240-241-242-243-244-245-246-247-248-249-250-251-252-253-254-255-256-257-258-259-260-261-262-263-264-265-266-267-268-269-270-271-272-273-274-275-276-277-278-279-280-281-282-283-284-285-286-287-288-289-290-291-292-293-294-295-296-297-298-299-300-301-302-303-304-305-306-307-308-309-310-311-312-313-314-315-316-317-318-319-320-321-322-323-324-325-326-327-328-329-330-331-332-333-334-335-336-337-338-339-340-341-342-343-344-345-346-347-348-349-350-351-352-353-354-355-356-357-358-359-360-361-362-363-364-365-366-367-368-369-370-371-372-373-374-375-376-377-378-379-380-381-382-383-384-385-386-387-388-389-390-391-392-393-394-395-396-397-398-399-400-401-402-403-404-405-406-407-408-409-410-411-412-413-414-415-416-417-418-419-420-421-422-423-424-425-426-427-428-429-430-431-432-433-434-435-436-437-438-439-440-441-442-443-444-445-446-447-448-449-450-451-452-453-454-455-456-457-458-459-460-461-462-463-464-465-466-467-468-469-470-471-472-473-474-475-476-477-478-479-480-481-482-483-484-485-486-487-488-489-490-491-492-493-494-495-496-497-498-499-500-501-502-503-504-505-506-507-508-509-510-511-512-513-514-515-516-517-518-519-520-521-522-523-524-525-526-527-528-529-530-531-532-533-534-535-536-537-538-539-540-541-542-543-544-545-546-547-548-549-550-551-552-553-554-555-556-557-558-559-560-561-562-563-564-565-566-567-568-569-570-571-572-573-574-575-576-577-578-579-580-581-582-583-584-585-586-587-588-589-590-591-592-593-594-595-596-597-598-599-600-601-602-603-604-605-606-607-608-609-610-611-612-613-614-615-616-617-618-619-620-621-622-623-624-625-626-627-628-629-630-631-632-633-634-635-636-637-638-639-640-641-642-643-644-645-646-647-648-649-650-651-652-653-654-655-656-657-658-659-660-661-662-663-664-665-666-667-668-669-670-671-672-673-674-675-676-677-678-679-680-681-682-683-684-685-686-687-688-689-690-691-692-693-694-695-696-697-698-699-700-701-702-703-704-705-706-707-708-709-710-711-712-713-714-715-716-717-718-719-720-721-722-723-724-725-726-727-728-729-730-731-732-733-734-735-736-737-738-739-740-741-742-743-744-745-746-747-748-749-750-751-752-753-754-755-756-757-758-759-760-761-762-763-764-765-766-767-768-769-770-771-772-773-774-775-776-777-778-779-780-781-782-783-784-785-786-787-788-789-790-791-792-793-794-795-796-797-798-799-800-801-802-803-804-805-806-807-808-809-810-811-812-813-814-815-816-817-818-819-820-821-822-823-824-825-826-827-828-829-830-831-832-833-834-835-836-837-838-839-840-841-842-843-844-845-846-847-848-849-850-851-852-853-854-855-856-857-858-859-860-861-862-863-864-865-866-867-868-869-870-871-872-873-874-875-876-877-878-879-880-881-882-883-884-885-886-887-888-889-890-891-892-893-894-895-896-897-898-899-900-901-902-903-904-905-906-907-908-909-910-911-912-913-914-915-916-917-918-919-920-921-922-923-924-925-926-927-928-929-930-931-932-933-934-935-936-937-938-939-940-941-942-943-944-945-946-947-948-949-950-951-952-953-954-955-956-957-958-959-960-961-962-963-964-965-966-967-968-969-970-971-972-973-974-975-976-977-978-979-980-981-982-983-984-985-986-987-988-989-990-991-992-993-994-995-996-997-998-999-1000-1001-1002-1003-1004-1005-1006-1007-1008-1009-1010-1011-1012-1013-1014-1015-1016-1017-1018-1019-1020-1021-1022-1023-1024-1025-1026-1027-1028-1029-1030-1031-1032-1033-1034-1035-1036-1037-1038-1039-1040-1041-1042-1043-1044-1045-1046-1047-1048-1049-1050-1051-1052-1053-1054-1055-1056-1057-1058-1059-1060-1061-1062-1063-1064-1065-1066-1067-1068-1069-1070-1071-1072-1073-1074-1075-1076-1077-1078-1079-1080-1081-1082-1083-1084-1085-1086-1087-1088-1089-1090-1091-1092-1093-1094-1095-1096-1097-1098-1099-1100-1101-1102-1103-1104-1105-1106-1107-1108-1109-1110-1111-1112-1113-1114-1115-1116-1117-1118-1119-1120-1121-1122-1123-1124-1125-1126-1127-1128-1129-1130-1131-1132-1133-1134-1135-1136-1137-1138-1139-1140-1141-1142-1143-1144-1145-1146-1147-1148-1149-1150-1151-1152-1153-1154-1155-1156-1157-1158-1159-1160-1161-1162-1163-1164-1165-1166-1167-1168-1169-1170-1171-1172-1173-1174-1175-1176-1177-1178-1179-1180-1181-1182-1183-1184-1185-1186-1187-1188-1189-1190-1191-1192-1193-1194-1195-1196-1197-1198-1199-1200-1201-1202-1203-1204-1205-1206-1207-1208-1209-1210-1211-1212-1213-1214-1215-1216-1217-1218-1219-1220-1221-1222-1223-1224-1225-1226-1227-1228-1229-1230-1231-1232-1233-1234-1235-1236-1237-1238-1239-1240-1241-1242-1243-1244-1245-1246-1247-1248-1249-1250-1251-1252-1253-1254-1255-1256-1257-1258-1259-1260-1261-1262-1263-1264-1265-1266-1267-1268-1269-1270-1271-1272-1273-1274-1275-1276-1277-1278-1279-1280-1281-1282-1283-1284-1285-1286-1287-1288-1289-1290-1291-1292-1293-1294-1295-1296-1297-1298-1299-1300-1301-1302-1303-1304-1305-1306-1307-1308-1309-1310-1311-1312-1313-1314-1315-1316-1317-1318-1319-1320-1321-1322-1323-1324-1325-1326-1327-1328-1329-1330-1331-1332-1333-1334-1335-1336-1337-1338-1339-1340-1341-1342-1343-1344-1345-1346-1347-1348-1349-1350-1351-1352-1353-1354-1355-1356-1357-1358-1359-1360-1361-1362-1363-1364-1365-1366-1367-1368-1369-1370-1371-1372-1373-1374-1375-1376-1377-1378-1379-1380-1381-1382-1383-1384-1385-1386-1387-1388-1389-1390-1391-1392-1393-1394-1395-1396-1397-1398-1399-1400-1401-1402-1403-1404-1405-1406-1407-1408-1409-1410-1411-1412-1413-1414-1415-1416-1417-1418-1419-1420-1421-1422-1423-1424-1425-1426-1427-1428-1429-1430-1431-1432-1433-1434-1435-1436-1437-1438-1439-1440-1441-1442-1443-1444-1445-1446-1447-1448-1449-1450-1451-1452-1453-1454-1455-1456-1457-1458-1459-1460-1461-1462-1463-1464-1465-1466-1467-1468-1469-1470-1471-1472-1473-1474-1475-1476-1477-1478-1479-1480-1481-1482-1483-1484-1485-1486-1487-1488-1489-1490-1491-1492-1493-1494-1495-1496-1497-1498-1499-1500-1501-1502-1503-1504-1505-1506-1507-1508-1509-1510-1511-1512-1513-1514-1515-1516-1517-1518-1519-1520-1521-1522-1523-1524-1525-1526-1527-1528-1529-1530-1531-1532-1533-1534-1535-1536-1537-1538-1539-1540-1541-1542-1543-1544-1545-1546-1547-1548-1549-1550-1551-1552-1553-1554-1555-1556-1557-1558-1559-1560-1561-1562-1563-1564-1565-1566-1567-1568-1569-1570-1571-1572-1573-1574-1575-1576-1577-1578-1579-1580-1581-1582-1583-1584-1585-1586-1587-1588-1589-1590-1591-1592-1593-1594-1595-1596-1597-1598-1599-1600-1601-1602-1603-1604-1605-1606-1607-1608-1609-1610-1611-1612-1613-1614-1615-1616-1617-1618-1619-1620-1621-1622-1623-1624-1625-1626-1627-1628-1629-1630-1631-1632-1633-1634-1635-1636-1637-1638-1639-1640-1641-1642-1643-1644-1645-1646-1647-1648-1649-1650-1651-1652-1653-1654-1655-1656-1657-1658-1659-1660-1661-1662-1663-1664-1665-1666-1667-1668-1669-1670-1671-1672-1673-1674-1675-1676-1677-1678-1679-1680-1681-1682-1683-1684-1685-1686-1687-1688-1689-1690-1691-1692-1693-1694-1695-1696-1697-1698-1699-1700-1701-1702-1703-1704-1705-1706-1707-1708-1709-1710-1711-1712-1713-1714-1715-1716-1717-1718-1719-1720-1721-1722-1723-1724-1725-1726-1727-1728-1729-1730-1731-1732-1733-1734-1735-1736-1737-1738-1739-1740-1741-1742-1743-1744-1745-1746-1747-1748-1749-1750-1751-1752-1753-1754-1755-1756-1757-1758-1759-1760-1761-1762-1763-1764-1765-1766-1767-1768-1769-1770-1771-1772-1773-1774-1775-1776-1777-1778-1779-1780-1781-1782-1783-1784-1785-1786-1787-1788-1789-1790-1791-1792-1793-1794-1795-1796-1797-1798-1799-1800-1801-1802-1803-1804-1805-1806-1807-1808-1809-1810-1811-1812-1813-1814-1815-1816-1817-1818-1819-1820-1821-1822-1823-1824-1825-1826-1827-1828-1829-1830-1831-1832-1833-1834-1835-1836-1837-1838-1839-1840-1841-1842-1843-1844-1845-1846-1847-1848-1849-1850-1851-1852-1853-1854-1855-1856-1857-1858-1859-1860-1861-1862-1863-1864-1865-1866-1867-1868-1869-1870-1871-1872-1873-1874-1875-1876-1877-1878-1879-1880-1881-1882-1883-1884-1885-1886-1887-1888-1889-1890-1891-1892-1893-1894-1895-1896-1897-1898-1899-1900-1901-1902-1903-1904-1905-1906-1907-1908-1909-1910-1911-1912-1913-1914-1915-1916-1917-1918-1919-1920-1921-1922-1923-1924-1925-1926-1927-1928-1929-1930-1931-1932-1933-1934-1935-1936-1937-1938-1939-1940-1941-1942-1943-1944-1945-1946-1947-1948-1949-1950-1951-1952-1953-1954-1955-1956-1957-1958-1959-1960-1961-1962-1963-1964-1965-1966-1967-1968-1969-1970-1971-1972-1973-1974-1975-1976-1977-1978-1979-1980-1981-1982-1983-1984-1985-1986-1987-1988-1989-1990-1991-1992-1993-1994-1995-1996-1997-1998-1999-2000-2001-2002-2003-2004-2005-2006-2007-2008-2009-2010-2011-2012-2013-2014-2015-2016-2017-2018-2019-2020-2021-2022-2023-2024-2025-2026-2027-2028-2029-2030-2031-2032-2033-2034-2035-2036-2037-2038-2039-2040-2041-2042-2043-2044-2045-2046-2047-2048-2049-2050-2051-2052-2053-2054-2055-2056-2057-2058-2059-2060-2061-2062-2063-2064-2065-2066-2067-2068-2069-2070-2071-2072-2073-2074-2075-2076-2077-2078-2079-2080-2081-2082-2083-2084-2085-2086-2087-2088-2089-2090-2091-2092-2093-2094-2095-2096-2097-2098-2099-2100-2101-2102-2103-2104-2105-2106-2107-2108-2109-2110-2111-2112-2113-2114-2115-2116-2117-2118-2119-2120-2121-2122-2123-2124-2125-2126-2127-2128-2129-2130-2131-2132-2133-2134-2135-2136-2137-2138-2139-2140-2141-2142-2143-2144-2145-2146-2147-2148-2149-2150-2151-2152-2153-2154-2155-2156-2157-2158-2159-2160-2161-2162-2163-2164-2165-2166-2167-2168-2169-2170-2171-2172-2173-2174-2175-2176-2177-2178-2179-2180-2181-2182-2183-2184-2185-2186-2187-2188-2189-2190-2191-2192-2193-2194-2195-2196-2197-2198-2199-2200-2201-2202-2203-2204-2205-2206-2207-2208-2209-2210-2211-2212-2213-2214-2215-2216-2217-2218-2219-2220-2221-2222-2223-2224-2225-2226-2227-2228-2229-2230-2231-2232-2233-2234-2235-2236-2237-2238-2239-2240-2241-2242-2243-2244-2245-2246-2247-2248-2249-2250-2251-2252-2253-2254-2255-2256-2257-2258-2259-2260-2261-2262-2263-2264-2265-2266-2267-2268-2269-2270-2271-2272-2273-2274-2275-2276-2277-2278-2279-2280-2281-2282-2283-2284-2285-2286-2287-2288-2289-2290-2291-2292-2293-2294-2295-2296-2297-2298-2299-2300-2301-2302-2303-2304-2305-2306-2307-2308-2309-2310-2311-2312-2313-2314-2315-2316-2317-2318-2319-2320-2321-2322-2323-2324-2325-2326-2327-2328-2329-2330-2331-2332-2333-2334-2335-2336-2337-2338-2339-2340-2341-2342-2343-2344-2345-2346-2347-2348-2349-2350-2351-2352-2353-2354-2355-2356-2357-2358-2359-2360-2361-2362-2363-2364-2365-2366-2367-2368-2369-2370-2371-2372-2373-2374-2375-2376-2377-2378-2379-2380-2381-2382-2383-2384-2385-2386-2387-2388-2389-2390-2391-2392-2393-2394-2395-2396-2397-2398-2399-2400-2401-2402-2403-2404-2405-2406-2407-2408-2409-2410-2411-2412-2413-2414-2415-2416-2417-2418-2419-2420-2421-2422-2423-2424-2425-2426-2427-2428-2429-2430-2431-2432-2433-2434-2435-2436-2437-2438-2439-2440-2441-2442-2443-2444-2445-2446-2447-2448-2449-2450-2451-2452-2453-2454-2455-2456-2457-2458-2459-2460-2461-2462-2463-2464-2465-2466-2467-2468-2469-2470-2471-2472-2473-2474-2475-2476-2477-2478-2479-2480-2481-2482-2483-2484-2485-2486-2487-2488-2489-2490-2491-2492-2493-2494-2495-2496-2497-2498-2499-2500-2501-2502-2503-2504-2505-2506-2507-2508-2509-2510-2511-2512-2513-2514-2515-2516-2517-2518-2519-2520-2521-2522-2523-2524-2525-2526-2527-2528-2529-2530-2531-2532-2533-2534-2535-2536-2537-253

A4. Optimización de conexiones HTTPS y cifrado en el balanceador

Implementación:

```
http {
    # por defecto es round-robin
    server 192.168.10.2;
    server 192.168.10.3;
    server 192.168.10.4;
    server 192.168.10.5;
    server 192.168.10.6;
    server 192.168.10.7;
    server 192.168.10.8;
    server 192.168.10.9;
}
server {
    listen 80;
    listen 443 ssl http2;
    server_name nginx_balanceador;
    root /usr/share/nginx/html;
    index index.php index.html;

    ssl_protocols TLSv1.2 TLSv1.3;
    ssl_ciphers 'ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305';
    ssl_prefer_server_ciphers on;

    ssl_certificate /etc/nginx/ssl/Webs/servidor-web-full.crt;
    ssl_certificate_key /etc/nginx/ssl/Webs/servidor-web.key;
    ssl_trusted_certificate /etc/nginx/ssl/CA/ca-raiz.crt;

    ssl_session_cache shared:SSL:10m;
    ssl_session_timeout 20m;
}
```

Vemos que funciona correctamente:

```
> curl -k -I --http2 https://localhost
HTTP/2 200
server: nginx/1.27.5
date: Tue, 29 Apr 2025 20:43:01 GMT
content-type: text/html; charset=UTF-8
```

Comprobamos el cifrado que se usa con TLSv1.3:

```
> openssl s_client -connect localhost:443 -tls1_3 | grep "Cipher"
Can't use SSL_get_servername
depth=0 C = ES, ST = Granada, L = Granada, O = Universidad de Granada, OU = Universidad de Granada, CN = Florin Emanuel Todor Gliga, emailAddress = flotodor@correo.ugr.es
verify error:num=18:self-signed certificate
verify return:1
depth=0 C = ES, ST = Granada, L = Granada, O = Universidad de Granada, OU = Universidad de Granada, CN = Florin Emanuel Todor Gliga, emailAddress = flotodor@correo.ugr.es
verify return:1
New, TLSv1.3, Cipher is TLS_AES_256_GCM_SHA384
```

Para TLSv1.2:

```
> openssl s_client -connect localhost:443 -tls1_2 | grep "Cipher"
Can't use SSL_get_servername
depth=0 C = ES, ST = Granada, L = Granada, O = Universidad de Granada, OU = Universidad de Granada, CN = Florin Emanuel Todor Gliga, emailAddress = flotodor@correo.ugr.es
verify error:num=18:self-signed certificate
verify return:1
depth=0 C = ES, ST = Granada, L = Granada, O = Universidad de Granada, OU = Universidad de Granada, CN = Florin Emanuel Todor Gliga, emailAddress = flotodor@correo.ugr.es
verify return:1
New, TLSv1.2, Cipher is ECDHE-RSA-AES256-GCM-SHA384
Cipher      : ECDHE-RSA-AES256-GCM-SHA384
```

Comprobamos que no se utilizan los protocolos TLSv1.1 ni 1.0:

```
Cipher      : ECDHE-RSA-AES256-GCM-SHA384
> openssl s_client -connect localhost:443 -tls1_1 | grep "Cipher"
4067E30587750000:error:0A0000BF:SSL routines:tls_setup_handshake:no protocols available:../ssl/statem/statem_lib.c:104:
New, (NONE), Cipher is (NONE)
> openssl s_client -connect localhost:443 -tls1_0 | grep "Cipher"
s_client: Unknown option: -tls1_0
s_client: Use -help for summary.
```

3. Uso de Inteligencia Artificial Generativa

Durante la generación de esta documentación he comentado el uso de la IA en algunas de las partes.

Enlace al chat: <https://chatgpt.com/c/680aae55-8504-800d-8786-07955235a4b3>

Como he comentado en todas las prácticas hasta ahora, es una buena herramienta y la verdad es que me ha ayudado debido a no poder asistir a la explicación de la práctica.

Me ha ayudado con la generación de los certificados, tanto para la parte básica como avanzada. Me ha explicado la utilidad de la cadena de verificación con la generación del certificado CA, SubCA, y de las webs.

En sí, el código generado por la IA no me ha dado error, los errores que se me han producido en el desarrollo de la práctica han sido más por fallos técnicos leves (no haber enviado bien los ficheros, no modificar los ficheros correctos, etc) por mi parte pero solucionados posteriormente.

Por otro lado, me ha comentado curiosidades sobre los cifrados, protocolos de seguridad, el uso de los tickets de sesión que no los conocía.

He tenido alguna discusión con GPT respecto a la generación del certificado para las webs de las tareas básicas. Esto aunque GPT no se dio cuenta al final era porque había creado yo los certificados con el usuario sudo en vez de mi usuario florin. Esto ha influido en tener que solucionar un error debido a que se copiaba el certificado a las webs pero con 0 bytes de datos. GPT no me daba solución y al final me di cuenta que era ese error "tonto".

Por lo demás todo ha ido perfectamente con la ayuda de la IA.