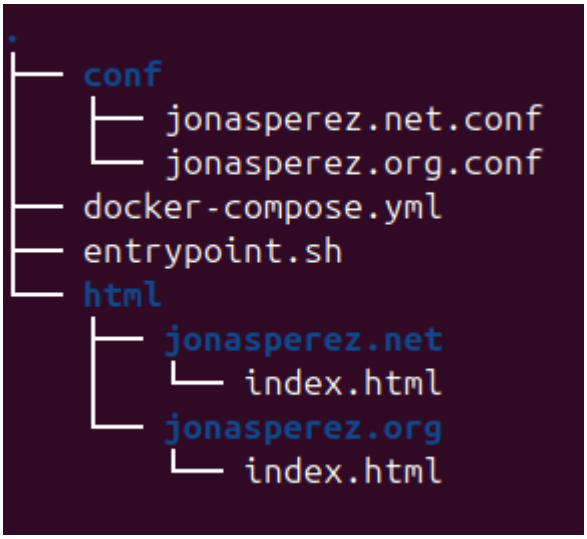
PRÁCTICA SERVIDORES WEB: SERVIDOR WEB APACHE + SSL SOBRE DOCKER

En base a esta estructura:



Vamos a implementar un cifrado SSL con el objetivo de cifrar el contenido y que nuestra página sea segura. De modo que, el primer paso será ejecutar el siguiente comando en el directorio:

mkdir certs

openssl req -x509 -nodes -days 365 -newkey rsa:2048 \

- keyout ./certs/apache-selfsigned.key \
- out ./certs/apache-selfsigned.crt \
- subj "/C=ES/ST=Valencia/L=Cheste/O=CIPFPCheste/OU=IT/CN=jonasperez.net"

En este caso estamos empezando en .net, luego haremos lo mismo en .org.

El siguiente paso será modificar nuestro docker-compose.yml, donde mapearemos el puerto 443 y montaremos la nueva carpeta de certificados:nano

```
  GNU nano 7.2                          docker-compose.yml
services:
  apache:
    image: httpd:2.4
    container_name: apache-multisite
    ports:
      - "8080:80"
      - "443:443"
    volumes:
      - ./html:/var/www/
      - ./conf:/usr/local/apache2/conf/vhosts/
      - ./certs:/etc/apache2/certs
      - ./entrypoint.sh:/entrypoint.sh
    entrypoint: ["/bin/bash", "/entrypoint.sh"]
    restart: unless-stopped
```

Seguido de esto, debemos modificar el archivo jonasperez.net.conf:

```
  GNU nano 7.2                          jonasperez.net.conf
<VirtualHost *:80>
    ServerName jonasperez.net
    Redirect permanent / https://jonasperez.net/
</VirtualHost>

<VirtualHost *:443>
    ServerName jonasperez.net
    DocumentRoot /var/www/jonasperez.net

    SSLEngine on
    SSLCertificateFile /etc/apache2/certs/apache-selfsigned.crt
    SSLCertificateKeyFile /etc/apache2/certs/apache-selfsigned.key

    <Directory /var/www/jonasperez.net>
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

Ahora repetiremos este proceso, pero con el .org:

mkdir certs

openssl req -x509 -nodes -days 365 -newkey rsa:2048 \

-keyout ./certs/apache-selfsigned.key \

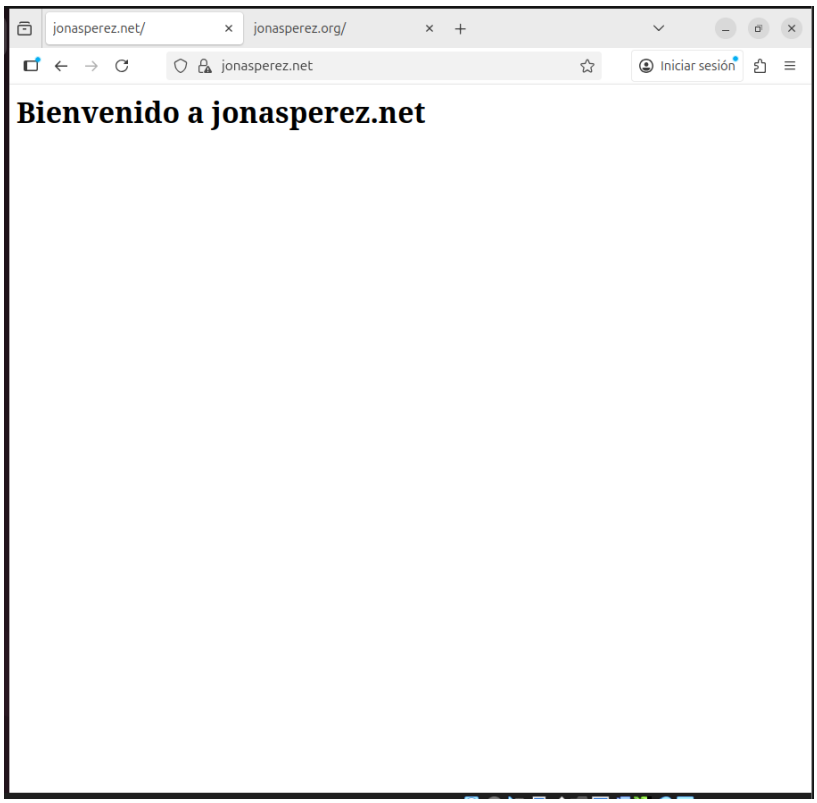
-out ./certs/apache-selfsigned.crt \

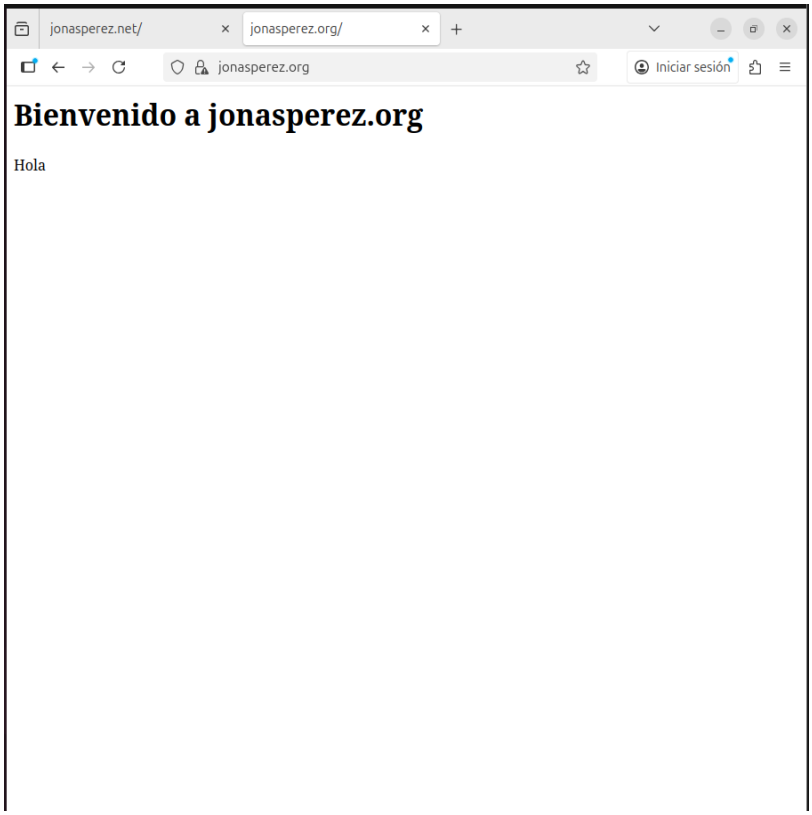
-subj "/C=ES/ST=Valencia/L=Cheste/O=CIPFPCheste/OU=IT/CN=jonasperez.org"

Por último, vamos a actualizar el archivo entrypoint.sh:

```
#!/bin/bash

mkdir -p /usr/local/apache2/conf/vhosts

# Incluir vhosts

if ! grep -q "Include /usr/local/apache2/conf/vhosts/*.conf"
/usr/local/apache2/conf/httpd.conf; then

echo "Include /usr/local/apache2/conf/vhosts/*.conf" >>
/usr/local/apache2/conf/httpd.conf

fi

# Activar módulos necesarios (SSL, rewrite, headers)

sed -i 's/#LoadModule ssl_module/LoadModule ssl_module/'
/usr/local/apache2/conf/httpd.conf

sed -i 's/#LoadModule rewrite_module/LoadModule rewrite_module/'
/usr/local/apache2/conf/httpd.conf

sed -i 's/#LoadModule headers_module/LoadModule headers_module/'
/usr/local/apache2/conf/httpd.conf

sed -i 's/#LoadModule socache_shmcb_module/LoadModule socache_shmcb_module/'
/usr/local/apache2/conf/httpd.conf

# Asegurarse de que Apache escuche en el 443 si no está configurado

if ! grep -q "Listen 443" /usr/local/apache2/conf/httpd.conf; then

echo "Listen 443" >> /usr/local/apache2/conf/httpd.conf

fi

httpd-foreground
```

Y ahora vamos al navegador e intentaremos entrar con https://jonasperez.net y https://jonasperez.org:





Cuando entramos, el navegador nos da un aviso de conexión no segura, ya que es un certificado auto-firmado, pero al darle en configuración avanzada y acceder a jonasperez.net/org, entraremos al sitio.