# Segunda practica arquitectura en la nube

Luis Peregrina Ojeda 2ºASIR Caja Mágica

Como siempre actualizamos los paquetes por si acaso

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo apt update && sudo apt upgrade -y
[sudo] password for luis:
Hit:1 http://archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
```

Después comprobamos que tenemos apache2 instalado (si lo hacemos en la misma maquina ya lo tenemos por la practica anterior)

```
is@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo apt install apache2 -y
ading package lists... Done
ilding dependency tree... Done
ading state information... Done
```

Y luego iremos al archivo "/etc/apache2/ports.conf" que como su nombre indica nos ayudara a cambiar los puertos en este caso el de escucha

(Modificamos el primero "Listen 80" > "Listen 8080")

```
 GNU nano 7.2                                    /etc/
# If you just change the port or add more ports
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 8080

<IfModule ssl_module>
        Listen 443
</IfModule>

<IfModule mod_gnutls.c>
        Listen 443
</IfModule>
```

Despues ejecutaremos este comando "sudo nano /etc/apache2/sites-available/000-default.conf" y haremos exactamente lo mismo con esta línea de comando

```
 GNU nano 7.2
<VirtualHost *:8080>
        # The ServerName di
```

Probaremos que tenemos instalado el php con el comando “sudo apt install php libapache2-mod-php -y” y despues reiniciaremos el sistema y comprobaremos su estado “sudo systemctl restart apache2” y cambiaremos el restart por status y comprobamos el estado.

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo apt install php libapache2-mod-php -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
php is already the newest version (2:8.3+93ubuntu2).
libapache2-mod-php is already the newest version (2:8.3+93ubuntu2).
The following package was automatically installed and is no longer required:
  libllvm19
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo systemctl restart apache2
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Thu 2025-10-09 14:22:45 CEST; 8s ago
```

Haremos un “echo "<?php phpinfo ();?>" | sudo tee /var/www/html/info.php” y nos saldra que el archivo esta creado

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ echo "<?php phpinfo(); ?>" | sudo tee /var/www/ht
ml/info.php
<?php phpinfo(); ?>
```

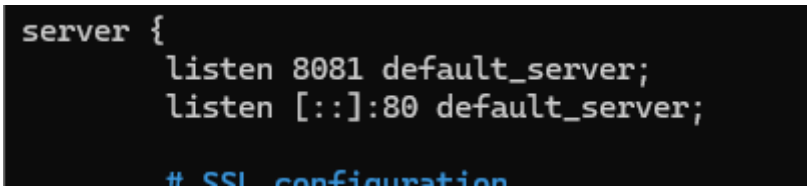
Despues probamos curl con http://localhost:8080/info.php

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ curl http://localhost:8080/info.php
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"><head>
<style type="text/css">
body {background-color: #fff; color: #222; font-family: sans-serif;}
pre {margin: 0; font-family: monospace;}
a:link {color: #009; text-decoration: none; background-color: #fff;}
a:hover {text-decoration: underline;}
table {border-collapse: collapse; border: 0; width: 934px; box-shadow: 1px 2px 3px rgba(0, 0, 0, 0.2)
}
.center {text-align: center;}
.center table {margin: 1em auto; text-align: left;}
.center th {text-align: center !important;}
td, th {border: 1px solid #666; font-size: 75%; vertical-align: baseline; padding: 4px 5px;}
th {position: sticky; top: 0; background: inherit;}
h1 {font-size: 150%;}
h2 {font-size: 125%;}
h2 a:link, h2 a:visited{color: inherit; background: inherit;}
.p {text-align: left;}
.e {background-color: #ccf; width: 300px; font-weight: bold;}
.h {background-color: #99c; font-weight: bold;}
.v {background-color: #ddd; max-width: 300px; overflow-x: auto; word-wrap: break-word;}
.v i {color: #999;}
img {float: right; border: 0;}
hr {width: 934px; background-color: #ccc; border: 0; height: 1px;}
:root {--php-dark-grey: #333; --php-dark-blue: #4F5B93; --php-medium-blue: #8892BF; --php-light-blue:
#E2E4EF; --php-accent-purple: #793862}@media (prefers-color-scheme: dark) {
  body {background: var(--php-dark-grey); color: var(--php-light-blue)}
  .h td, td.e, th {border-color: #606A90}
  td {border-color: #505153}
  .e {background-color: #404A77}
  .h {background-color: var(--php-dark-blue)}
  .v {background-color: var(--php-dark-grey)}
  hr {background-color: #505153}
}
</style>
```

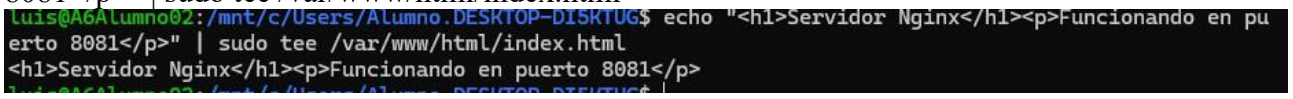
Ahora empezamos la segunda practica en Nginx. (deberíamos tenerlo instalado de antes y solo deberíamos cambiar los puertos)

Sudo apt install nginx -y

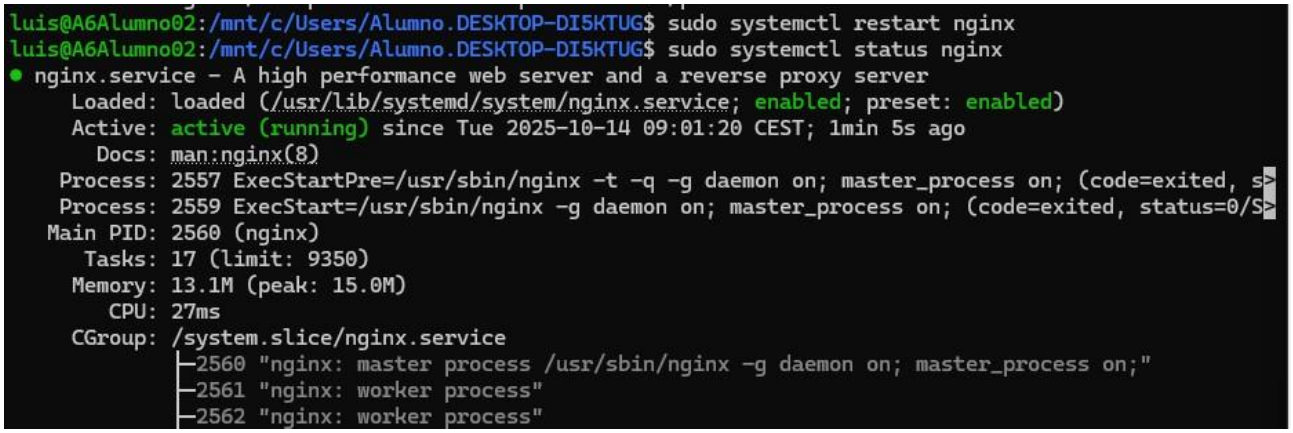
Despues vamos a “sudo nano /etc/nginx/sites-available/default” y cambiamos el puerto “Listen 80” a “Listen 8081”.




Y ahora crearemos el archivo con “echo "<h1>Servidor Nginx</h1> Funcionando en puerto 8081 " | sudo tee /var/www/html/index.html”

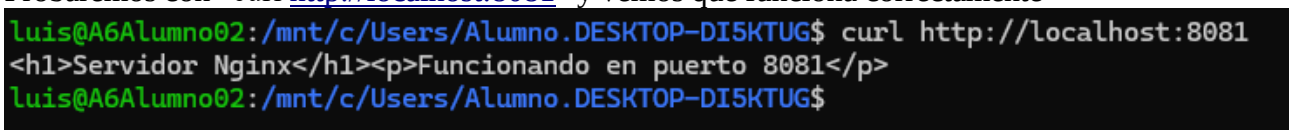



Haremos un reinicio del sistema para comprobar que todos los cambios funcionen correctamente y luego probaremos con el status




s

Probaremos con “curl http://localhost:8081” y vemos que funciona correctamente




Ahora empezaremos con la instalación de caddy con “sudo apt install -y debian-keyring debian-archive-keyring apt-transport-https curl”

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo apt install -y debian-keyring debian-arch
-keyring apt-transport-https curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apt-transport-https is already the newest version (2.8.3).
curl is already the newest version (8.5.0-2ubuntu10.6).
The following package was automatically installed and is no longer required:
  libllvm19
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  debian-archive-keyring debian-keyring
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 31.5 MB of archives.
After this operation, 33.4 MB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu noble/universe amd64 debian-archive-keyring all 2023.4ubuntu
```

Agregamos los repositorios caddy al sistema con “curl -1sLf 'https://dl.cloudsmith.io/public/caddy /stable/gpg.key' | sudo gpg --dearmor -o /usr/share/keyrings/caddy-stable-archive-keyring.gpg” y “curl -1sLf 'https://dl.cloudsmith.io/public/caddy /stable/debian.deb.txt' | sudo tee /etc/apt/sources.list.d/caddy-stable.list”

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ curl -1sLf 'https://dl.cloudsmith.io/public/caddy
/stable/gpg.key' | sudo gpg --dearmor -o /usr/share/keyrings/caddy-stable-archive-keyring.gpg
File '/usr/share/keyrings/caddy-stable-archive-keyring.gpg' exists. Overwrite? (y/N) y
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ curl -1sLf 'https://dl.cloudsmith.io/public/caddy
/stable/debian.deb.txt' | sudo tee /etc/apt/sources.list.d/caddy-stable.list
# Source: Caddy
# Site: https://github.com/caddyserver/caddy
# Repository: Caddy / stable
# Description: Fast, multi-platform web server with automatic HTTPS


deb [signed-by=/usr/share/keyrings/caddy-stable-archive-keyring.gpg] https://dl.cloudsmith.io/public/c
addy/stable/deb/debian any-version main

deb-src [signed-by=/usr/share/keyrings/caddy-stable-archive-keyring.gpg] https://dl.cloudsmith.io/publ
ic/caddy/stable/deb/debian any-version main
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$
```

Ahora actualizamos caddy

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo apt update && sudo apt install caddy -y
Get:1 https://dl.cloudsmith.io/public/caddy/stable/deb/debian any-version InRelease [14.8 kB]
Hit:2 https://download.docker.com/linux/ubuntu noble InRelease
Hit:3 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:4 http://archive.ubuntu.com/ubuntu noble InRelease
Hit:5 http://archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:6 http://archive.ubuntu.com/ubuntu noble-backports InRelease
Get:7 https://dl.cloudsmith.io/public/caddy/stable/deb/debian any-version/main amd64 Packages [4329 B]
Fetched 19.1 kB in 1s (28.2 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
Reading package lists... Done
```

Creamos un directorio para caddy por si acaso “sudo mkdir -p /var/www/caddy”

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo mkdir -p /var/www/caddy
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$
```

Con caddy hacemos todos estos echo y creamos un archivo @Markdown

```
C:\Users\Alumno.DESKTOP-DI5KTUG>wsl
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ echo "#Bienvenido a Caddy" | sudo tee /var/www/ca
ddy/README.md
[sudo] password for luis:
#Bienvenido a Caddy
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ echo "" | sudo tee -a /var/www/caddy/README.md

luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ echo "Este servidor esta funcionando correctament
e." | sudo tee -a /var/www/caddy/README.MD
Este servidor esta funcionando correctamente.
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ echo "" | sudo tee -a /var/www/caddy/README.md

luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ echo "## Caracteristicas" | sudo tee -a /var/www/
caddy/README.md
## Caracteristicas
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ echo "- Servidor moderno" | sudo tee -a /var/www/
caddy/README.md
- Servidor moderno
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ echo "HTTPS automatico" | sudo tee -a /var/www/ca
ddy/README.md
HTTPS automatico
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ echo "Facil configuracion" | sudo tee -a /var/www
/caddy/README.md
Facil configuracion
```

Probamos con curl a descargar una imagen “curl -o /tmp/test-image.jpg "https://www.python.org/static/apple-touch-icon-144x144-precomposer.png” y despues movemos la imagen de sitio “sudo mv /tmp/test-image.jpg /var/www/caddy/test.jpg”

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ curl -o /tmp/test-image.jpg "https://www.python.o
rg/static/apple-touch-icon-144x144-precomposer.png"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   146  100   146    0     0    927      0 --:--:-- --:--:-- --:--:--   929
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo mv /tmp/test-image.jpg /var/www/caddy/test.j
pg
```

Caddy archivo conf “sudo nano /etc/caddy/Caddyfile” y modificamos el puerto por 8082, y escribimos estas líneas que no están comentadas.

```
# domain name.

:8082 {
        # Set this path to your site's directory.
        root * /var/www/caddy

        # Enable the static file server.
        file_server browse

        @markdown path *.md
        header @markdown Content-Type text/plain

        # Another common task is to set up a reverse proxy:
        # reverse_proxy localhost:8080

        # Or serve a PHP site through php-fpm:
        # php_fastcgi localhost:9000
}
```

Reiniciamos el servidor caddy y probamos que este activado

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo systemctl restart caddy
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo systemctl status caddy
● caddy.service - Caddy
     Loaded: loaded (/usr/lib/systemd/system/caddy.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-10-14 14:38:49 CEST; 17s ago
       Docs: https://caddyserver.com/docs/
   Main PID: 866 (caddy)
      Tasks: 11 (limit: 9350)
     Memory: 49.0M (peak: 49.9M)
        CPU: 75ms
     CGroup: /system.slice/caddy.service
             └─866 /usr/bin/caddy run --environ --config /etc/caddy/Caddyfile

Oct 14 14:38:49 A6Alumno02 caddy[866]: {"level":"info","ts":1760443529.9451227,"logger":"a
```

Y ahora probamos con curl “curl http://localhost:8082/”

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ curl http://localhost:8082/

<!DOCTYPE html>
<html>
        <head>
                <title>/</title>
                <link rel="canonical" href="//"  />
                <meta charset="utf-8">
                <meta name="color-scheme" content="light dark">
                <meta name="viewport" content="width=device-width, initial-scale=1.0">
<style nonce="7d18b24b-dc70-4f14-8289-6b206ed3bc3a">
* { padding: 0; margin: 0; box-sizing: border-box; }

body {
        font-family: Inter, system-ui, sans-serif;
        font-size: 16px;
        text-rendering: optimizespeed;
        background-color: #f3f6f7;
        min-height: 100vh;
}

img,
svg {
        vertical-align: middle;
        z-index: 1;
}

img {
        max-width: 100%;
        max-height: 100%;
        border-radius: 5px;
}

td img {
        max-width: 1.5em;
        max-height: 2em;
        object-fit: cover;
}
```

Probamos otra vez curl, pero el readme “curl http://localhost:8082/README.md”

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ curl http://localhost:8082/README.md
#Bienvenido a Caddy


## Caracteristicas
- Servidor moderno
HTTPS automatico
Facil configuracion
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$
```

Ahora empezamos la instalación de certbot como siempre “sudo apt install certbot python3-certbot-apache -y”

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo apt install certbot python3-certbot-apache -
y
[sudo] password for luis:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  libllvm19
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  augeas-lenses libaugeas0 python3-acme python3-augeas python3-certbot python3-configargparse
  python3-icu python3-josepy python3-parsedatetime python3-rfc3339
Suggested packages:
  augeas-doc python-certbot-doc python3-certbot-nginx augeas-tools python-acme-doc
  python-certbot-apache-doc
The following NEW packages will be installed:
  augeas-lenses certbot libaugeas0 python3-acme python3-augeas python3-certbot
  python3-certbot-apache python3-configargparse python3-icu python3-josepy python3-parsedatetime
  python3-rfc3339
0 upgraded, 12 newly installed, 0 to remove and 0 not upgraded.
```

continuamos con certbot y pedimos un certificado auto firmado con “sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsignet.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]:ESP
String too long, must be at most 2 bytes long
Country Name (2 letter code) [AU]:ES
State or Province Name (full name) [Some-State]:Madrid
Locality Name (eg, city) []:Madrid
Organization Name (eg, company) [Internet Widgits Pty Ltd]:company
Organizational Unit Name (eg, section) []:section
Common Name (e.g. server FQDN or YOUR name) []:luis
```

Habilitamos modulo SSL en apache con “sudo a2enmod ssl”

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo a2enmod ssl
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 se
s.
To activate the new configuration, you need to run:
  systemctl restart apache2
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$
```

Creamos configuración en este archivo SSL “sudo nano /etc/apache2/sites-available/default-ssl.conf”

```
#   Enable/Disable SSL for this virtual host.
SSLEngine on

#   A self-signed (snakeoil) certificate can be created by installi
#   the ssl-cert package. See
#   /usr/share/doc/apache2/README.Debian.gz for more info.
#   If both key and certificate are stored in the same file, only t
#   SSLCertificateFile directive is needed.
SSLCertificateFile      /etc/ssl/certs/apache-selfsigned.crt
SSLCertificateKeyFile   /etc/ssl/private/apache-selfsigned.key
```

Cambiamos el puerto SSL para que se escuche “sudo nano /etc/apache2/ports.conf”

```
Listen 8080
Listen 8443

<IfModule ssl_module>
        Listen 443
</IfModule>

<IfModule mod_gnutls.c>
        Listen 443
</IfModule>
```

Modificamos el puerto de VirtualHost SSL

```
  GNU nano 7.2                  /etc/apache2/sites-available/default-ssl.conf *
<VirtualHost *:8443>
        ServerAdmin webmaster@localhost
```

Habilitamos sitio SSL

```
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo a2ensite default-ssl.conf
Enabling site default-ssl.
To activate the new configuration, you need to run:
  systemctl reload apache2
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo systemctl reload apache2
Job for apache2.service failed.
See "systemctl status apache2.service" and "journalctl -xeu apache2.service" for details.
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ systemctl status apache2.service
● apache2.service - The Apache HTTP Server
```

Reiniciamos el sistema de apache2 y probamos el curl

Antes de probar el curl, apache2 fallaba (realmente fallaba todo el rato). En resumen, al final cambia el ssl y lo volví a poner y se resolvió (estuve 2 horas). Utilice estos comandos

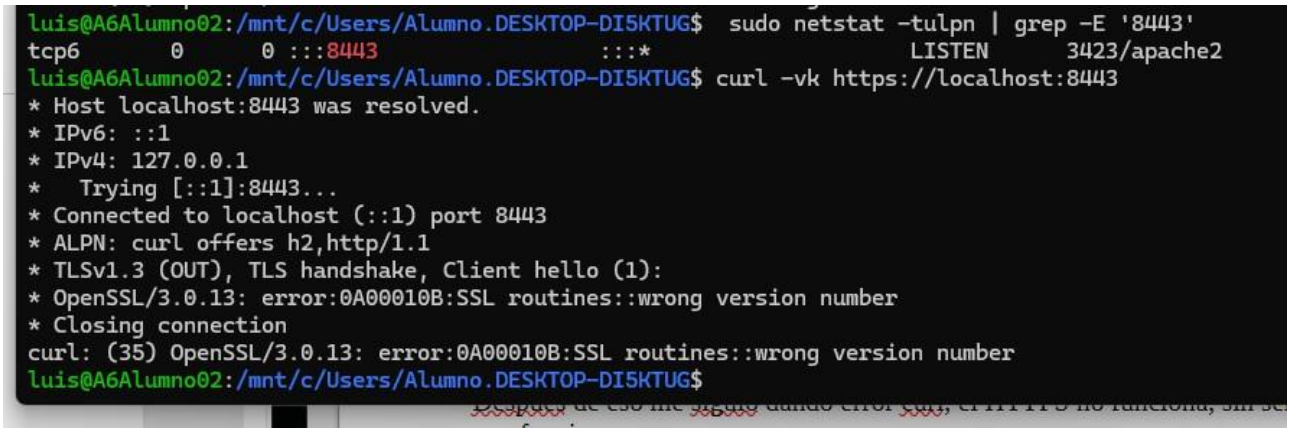
```
-----END: command not found
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ openssl rsa -in apache-selfsigned.key -out apache
-selfsigned.pem
Could not open file or uri for loading private key from apache-selfsigned.key
4097F33E94730000:error:16000069:STORE routines:ossl_store_get0_loader_int:unregistered scheme:../crypt
o/store/store_register.c:237:scheme=file
4097F33E94730000:error:80000002:system library:file_open:No such file or directory:../providers/implem
entations/storemgmt/file_store.c:267:calling stat(apache-selfsigned.key)
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ SSLCertificateFile /etc/ssl/certs/apache-selfsign
ed.crt
SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key
SSLCertificateFile: command not found
SSLCertificateKeyFile: command not found
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo apache2ctl configtest
Syntax OK
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo systemctl restart apache2
Job for apache2.service failed because the control process exited with error code.
See "systemctl status apache2.service" and "journalctl -xeu apache2.service" for details.
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$ sudo openssl req -new -newkey rsa:2048 -days 365
-nodes -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.csr
sudo openssl x509 -req -in /etc/ssl/certs/apache-selfsigned.csr -signkey /etc/ssl/private/apache-selfs
igned.key -out /etc/ssl/certs/apache-selfsigned.crt
Ignoring -days without -x509; not generating a certificate
.....+..+.......+...+..+....+..........+.+...+..+...+.+......+.............+....+.....+.........+...
```

Despues de eso me siguió dando error curl, el HTTPS no funciona, sin ser seguro, es decir, HTTP sí que funciona.

```
curl: (35) OpenSSL/3.0.13: error:0A00010B:SSL routines::wrong version number
luis@A6Alumno02:/mnt/c/Users/Alumno.DESKTOP-DI5KTUG$
```

Como no me sale esta parte de la practica voy a seguir haciendo el resto porque si no, no avanzo en ningún momento.

Despues de verificar el estado de nginx apache2 y caddy, probamos que el puerto se esté usando y que curl funcione (no funciona)




Finalmente, no conseguí que funcionase el curl, simplemente el puerto fallaba, al principio ni escuchaba el puerto 8443, luego lo escucho, pero sigue sin funcionar (sinceramente lo intente reinstalando wsl, probando en mi casa y cuando no era el curl eran los certificados SSL).