

Práctica de Servicios Web

Esta práctica es bastante pequeña, pero hay cosas en ella que van a ser preciso investigar ya sea con apuntes o con Google (o con Chat GPT que anda de moda).

1. Clientes HTTP

1.1. Navegadores en modo texto

Estamos acostumbrados a utilizar navegadores en entornos gráficos, pero existen navegadores en modo texto como Lynx.

Instala Lynx en Ubuntu y visita un par de páginas. Aporta capturas de pantalla.

1.2. Cabeceras de páginas

Visita una página web con un navegador y captura de alguna manera (hay muchas) las cabeceras que manda el cliente al servidor.

Aporta una captura de pantalla de una captura de las cabeceras, y un pequeño texto explicando para qué vale cada una. Comenta también el método que has utilizado para obtener las cabeceras.

1.3. Línea de comandos

Existen aplicaciones como **curl** que permiten realizar peticiones HTTP por línea de comandos (aunque también soportan otros protocolos como FTP).

Realiza y comenta 5 acciones diferentes que puedas realizar con **curl** y aporta capturas de pantalla de los resultados.

2. Servidor Apache

2.1. Instalación de Apache

Instala el servicio de Apache en una máquina virtual o en un contenedor de Docker.

Visita la página web de Apache por defecto y aporta una captura de pantalla.

2.2. Virtual Hosts

Los Virtual Hosts son un mecanismo que nos permiten que un servidor web pueda servir varias páginas web distintas.

- Crea una página web con tu nombre, un HTML básico que simplemente ponga tu nombre. Y sívela a través de un dominio tunombre.com.
- Sirve una página en un puerto diferente al 80 que simplemente ponga tu apellido. Sívela a través del nombre tuapellido.com.

En ambos casos, aporta captura de pantalla de un navegador visitando la página, y de los archivos de configuración empleados para cada virtualhost.

- Desactiva la página por defecto de Apache. ¿Qué página se sirve si se visita la IP del servidor? <http://127.0.0.1> (o la que tenga si se visita desde otra máquina).

2.3. Configura Apache para que use HTTPS

El protocolo HTTP sin cifrar es inseguro en aquellas páginas que requieren login, puesto que la información se transmite en texto plano.

Configura tu servidor Apache para que sirva la página con tunombre.com por HTTPS.

Aporta captura de pantalla.

2.4. Redirección de tráfico

Configura el servidor Apache para que redirija el tráfico de <http://tunombre.com> a <https://tunombre.com>.

Describe qué cambios has tenido que hacer para que esto suceda, y aporta captura de pantalla.

2.5. Página 404 personalizada

Configura tu Apache para que muestre una página 404 que hayas personalizado y aporta captura de pantalla.

2.6. Archivo .htaccess

Los archivos .htaccess permiten que los propios usuarios puedan adaptar las configuraciones de Apache a sus sitios web. A cambio, se pierde un poco de rendimiento, puesto que el servidor tendrá que acceder e interpretar estos ficheros.

Toda configuración que se pueda poner en un .htaccess se puede también poner en la configuración general del sitio.

Crea dos directorios dentro del directorio de la web que tienes con tunombre.com y pon varios archivos en ellos.

Crea un archivo .htaccess que impida que los archivos de un directorio se listen cuando se visitan a través de una URL. Por ejemplo, que al visitar <https://tunombre.com/directorio1> se muestren los archivos y al visitar <https://tunombre.com/directorio2> no se muestren.

2.7. Instalación de módulos

Instala el módulo de PHP en Apache y crea una página de prueba que demuestre que está funcionando.

3. Servidor nginx (Voluntaria)

El servidor Nginx es el servidor más utilizado actualmente en el mundo. Es más rápido a la hora de servir peticiones, pero no tan configurable como Apache.

3.1. Instala el servidor Nginx

Instala el servidor Nginx en otra máquina virtual.

3.2. Configura HTTPS en Nginx

Configura el uso de HTTP cifrado en el servidor Nginx.