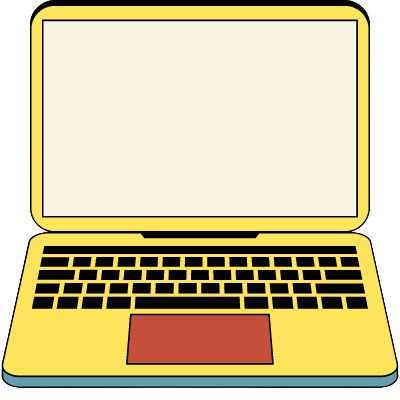
Configuración de Hosts Virtuales

**Christian Millán Soria**





- Curso: 2023-2024

- Clase: 2º DAW Tarde

- Profesor: David Hormigo Ramírez

- Módulo: Despliegue de Aplicaciones Web

- Fecha: 10/10/2023

**1. Configuración de hosts** 3

**1.1. Información sobre el alumno** 3

**1.2. Alan Turing** 8

**1.2.1. Imagen de Alan Turing** 9

**1.3. Página no encontrada** 12

**2. IP pública elástica** 14

**2.1. Creación** 14

**2.2. Asociación** 15

**2.3. Testeo** 17

**1. Configuración de hosts**

Para empezar, vamos a realizar la configuración de los hosts virtuales, que serán necesarios para que nuestro[**servidor Apache recién instalado🔗**](https://github.com/cmilsor245/daweb/blob/main/02-servidor_apache/Instalaci%C3%B3n%20y%20configuraci%C3%B3n%20b%C3%A1sica%20de%20un%20Servidor%20Web%20Apache.pdf) pueda ofrecer más de una web a la vez, que es como suelen funcionar en las aplicaciones reales.

El proceso requiere que obtengamos de nuevo la **dirección IP pública** de nuestra instancia en AWS y volvamos a descargar las **vockey**.

Nos conectaremos a la instancia utilizando [**PuTTy**](https://github.com/cmilsor245/daweb/blob/main/01-servidor_ubuntu_aws/Instalaci%C3%B3n%20de%20un%20servidor%20Ubuntu%20en%20AWS.pdf)/[**MobaXterm🔗**](https://github.com/cmilsor245/daweb/blob/main/01-servidor_ubuntu_aws/Instalaci%C3%B3n%20de%20un%20servidor%20Ubuntu%20en%20AWS.pdf).

La práctica se divide en 2 aspectos principales:

- **Información del alumno:** Se redirecciona a una página web donde se muestra información básica sobre el alumno.

- **Alan Turing:** Se redirecciona a una página donde se puede ver una imagen y un texto relacionados con [**Alan Turing**🔗](https://en.wikipedia.org/wiki/Alan_Turing).

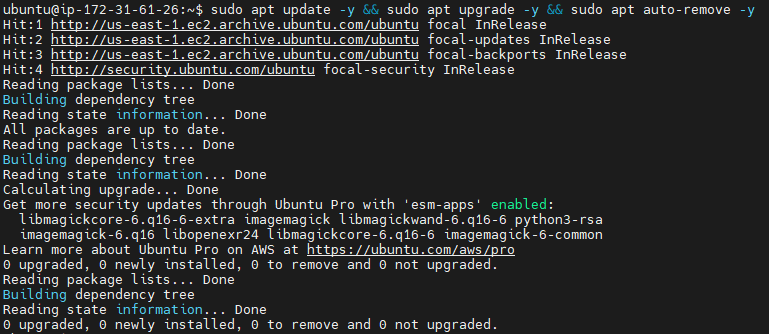
- **Página no encontrada:** Se redirecciona a una página por defecto, por si se ingresa una URL que no corresponde a las anteriores.

**1.1. Información sobre el alumno**

El objetivo de este apartado es que, al ingresar una de las siguientes URLs en un navegador web, se muestre siempre una página web en concreto, la cual mostrará información básica del alumno.

Para ello, lo primero que debemos hacer es crear un archivo de configuración del [**nuevo sitio de Apache**🔗](https://www.digitalocean.com/community/tutorials/how-to-set-up-apache-virtual-hosts-on-ubuntu-20-04).

Lo primero que debemos hacer siempre (recomendable) es actualizar la máquina con el comando “**sudo apt update -y && sudo apt upgrade -y && sudo apt auto-remove -y**”.



Ahora ejecutamos nuestro primer comando para el proceso como tal, el cual será “**sudo cp /etc/apache2/sites-available/misitio.conf /etc/apache2/sites-available/{nombre-alumno}.conf**”. Este comando se encargará de copiar el archivo de configuración del sitio que utilizamos en la práctica anterior en la misma ruta en la que se encuentra, pero con el nombre de cada alumno personalizado.

En mi caso el archivo se llamará “**christianmillan.conf**”.

Entramos en él para editarlo con el comando “**sudo nano /etc/apache2/sites-available/christianmillan.conf**”.

Una vez dentro, eliminamos absolutamente todo el contenido que este alberga ahora mismo.

Cuando hayamos hecho esto, podemos pasar a establecer el host virtual que hará acción para mostrar nuestra próxima página web.

Ingresaremos el siguiente contenido:

**<VirtualHost \*:80>**

**DocumentRoot /var/www/{nombre-alumno}/**

**ServerName www. {nombre-alumno}.com**

**ServerAlias {nombre-alumno}.com www.{nombre-alumno}.net www.{nombre-alumno}.es {nombre-alumno}.net {nombre-alumno}.es**

**</VirtualHost>**



Para comprobar que la configuración es válida podemos utilizar el comando “**apachectl configtest**”.



Para entender qué significa todo esto debemos tener claro lo siguiente: un [**virtual host**🔗](https://httpd.apache.org/docs/2.4/vhosts/) es un elemento de configuración de Apache que utilizaremos para generar un redireccionamiento personalizado hacia las páginas web que queramos dentro de nuestro servidor.

Como se puede ver en el texto mencionado anteriormente, aparecen las siguientes directrices:

- “**DocumentRoot”:** Se refiere al directorio padre que alojará la página o contenido que queremos mostrar.

- **“ServerName”:** Es la URL por defecto que se utilizará como main para visitar nuestro contenido.

- **“ServerAlias”:** Todos los elementos que se especifiquen dentro de esta directriz se utilizarán como URLs alternativas para poder seguir accediendo a la nuestra en caso de no ingresar la URL por defecto.

En mi caso, todos los “**{nombre-alumno}**” los sustituyo por el texto “**christianmillan**”.

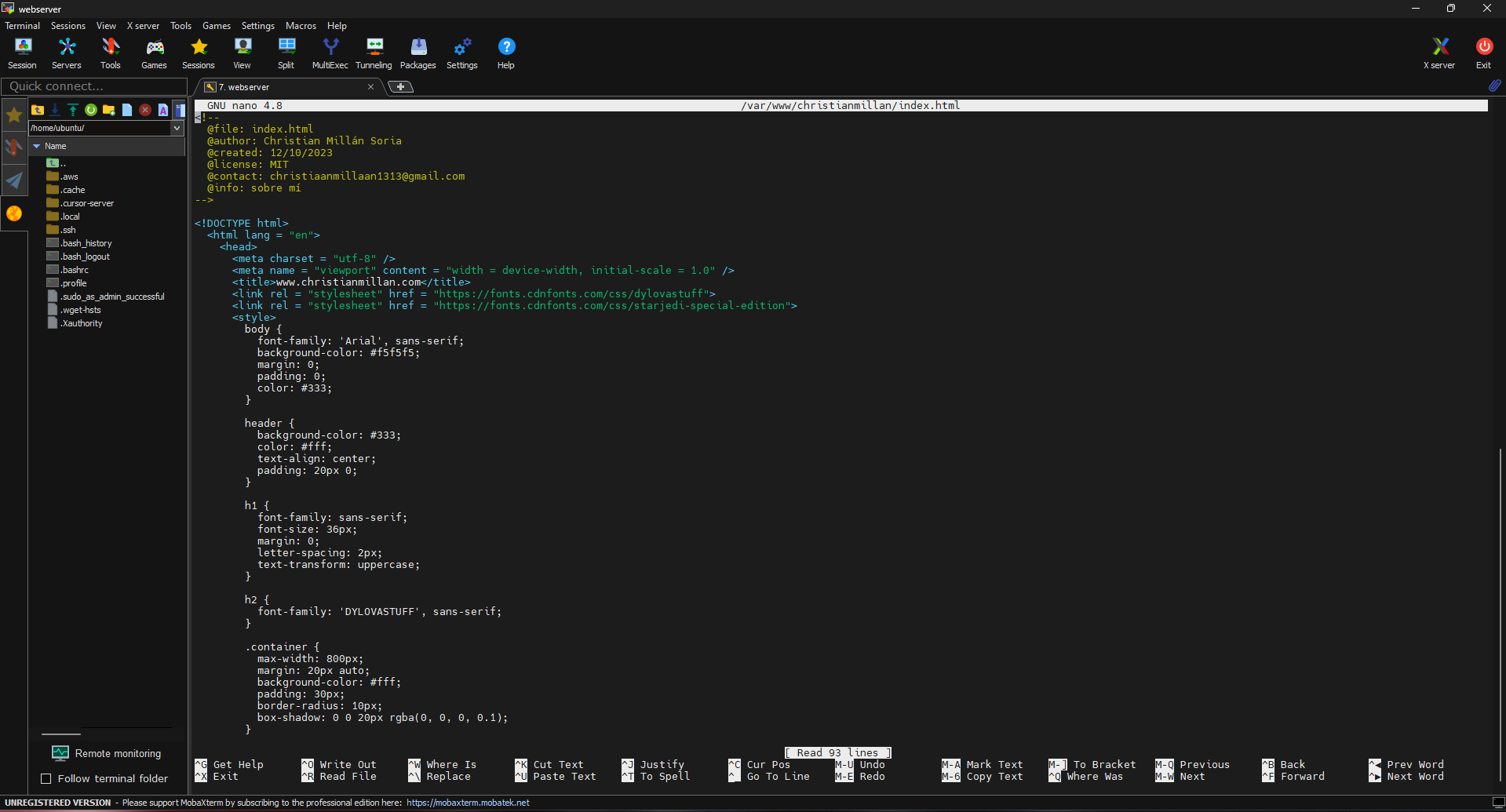
Una vez hecho esto, falta crear el archivo HTML principal que se mostrará en el navegador web.

Para hacer esto, utilizamos el comando “**sudo mkdir /var/www/{nombre-alumno}**”.

Esto ha creado el “DocumentRoot”, mencionado anteriormente.

Ahora creamos el archivo HTML con el comando “**sudo nano /var/www/{nombre-alumno}/index.html**”.

En mi caso he creado un código HTML básico creado por mí para mostrar de forma muy sencilla algo de información básica sobre mí.



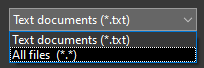
El último paso en la configuración de este primer apartado es habilitar el sitio que hemos creado y editar el archivo de hosts de nuestra máquina local para poder acceder al mismo.

Para lo primero ejecutamos el comando “**sudo a2ensite {nombre-alumno}.conf**”, seguido del comando “**sudo systemctl reload apache2**” para aplicar los cambios hechos en el servidor.

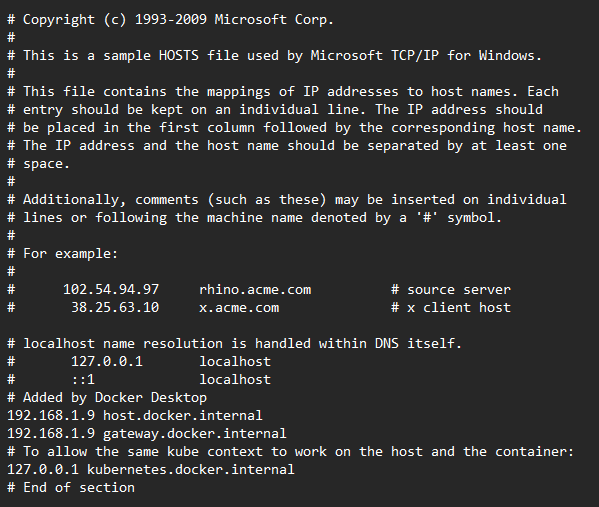
Para la segunda parte, debemos abrir el bloc de notas de Windows en **modo administrador**, ya que vamos a editar un archivo de la propia configuración de nuestro sistema operativo.

Esto se debe a que no contamos todavía con ningún servidor DNS que resuelva las URLs que ingresemos a ninguna dirección IP conocida ni tampoco con algún dominio comprado, por lo que es necesario configurar la máquina local para que convierta una URL personalizada en una dirección IP concreta.

Una vez dentro del bloc de notas, hacemos “**Ctrl + O**” para abrir un archivo situado en la ruta “**C:\Windows\System32\drivers\etc**” (suponiendo que el sistema operativo está instalado en la unidad de disco local C). El archivo es el llamado “**hosts**”. Si no aparece de primeras, debemos seleccionar la opción “**All files (\*.\*)**” situada en la esquina inferior derecha, ya que el archivo no tiene extensión.



Al abrirlo nos encontraremos con un contenido como el siguiente:



Nos dirigimos al final del documento, hacemos clic en la tecla “**Enter**” dos veces para separarnos del resto del contenido del archivo e ingresamos el siguiente texto:

**{dirección-ip-pública-instancia} www.{nombre-alumno}.com**

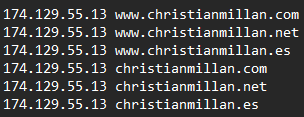
**{dirección-ip-pública-instancia} www.{nombre-alumno}.net**

**{dirección-ip-pública-instancia} www.{nombre-alumno}.es**

**{dirección-ip-pública-instancia} {nombre-alumno}.com**

**{dirección-ip-pública-instancia} {nombre-alumno}.net**

**{dirección-ip-pública-instancia} {nombre-alumno}.es**

****

Guardamos los cambios y salimos del archivo.

\* En Linux puede que el archivo sea diferente. Esta guía contempla el caso de realizar la práctica en Windows. [**Aquí🔗**](https://www.hostinger.com/tutorials/how-to-edit-hosts-file)hay una guía para realizar dicha tarea.

Una vez hecho todo esto, podemos abrir un navegador web y visitar cualquiera de las URls ingresadas en el archivo anterior y se nos resolverá la dirección a la que apunta nuestro nuevo sitio, en nuestra instancia en AWS.



**1.2. Alan Turing**

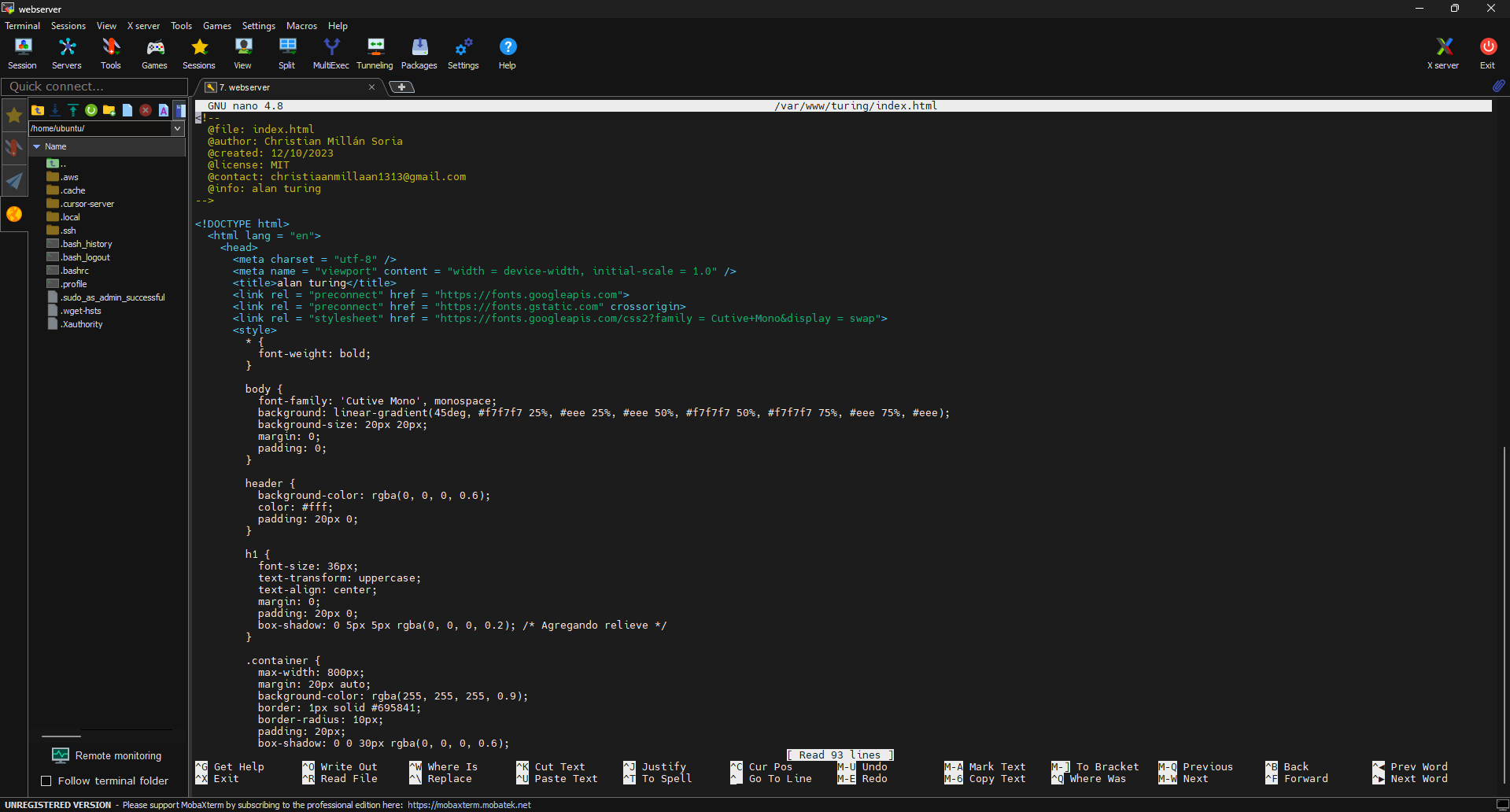
El siguiente paso de la práctica es crear un sitio personalizado sobre Alan Turing con una imagen del mismo alojada en el servidor.

Lo primero que hacemos es prepararnos el directorio del servidor donde vamos a guardar dicha imagen, el cual resultará ser un directorio por debajo del directorio que alojará la página web para este segundo sitio.

Con el comando “**sudo mkdir /var/www/turing**” creamos el directorio donde se encontrará nuestro HTML para la página de Alan Turing.

Dentro de esta carpeta crearemos una más llamada “**img**”: **sudo mkdir /var/www/turing/img**”. Aquí es donde se guardará la imagen.

Ya que estamos con este proceso de directorios vamos a crear el archivo HTML para la página con el comando “**sudo nano /var/www/turing/index.html**”, donde ingresaremos un código HTML que pueda mostrar la imagen (**debe ser la imagen alojada en el server, no una de Internet o de cualquier otra fuente**) y un texto relacionado con Alan Turing.



Creamos el archivo de configuración para el sitio de Turing.

Volvemos a utilizar el comando “cp” para copiar un archivo de configuración: “**sudo cp /etc/apache2/sites-available/{nombre-alumno}.conf /etc/apache2/sites-available/turing.conf**”.

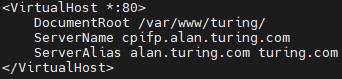
Entramos en él con el comando “nano”: “**sudo nano /etc/apache2/sites-available/turing.conf**”.

Cambiamos el contenido para que cuente con las siguientes directrices:

- “**turing**” como “**DocumentRoot**”.

- “**cpifp.alan.turing.com**” como “**ServerName**”.

- “**alan.turing.com turing.com**” como “**ServerAlias**”.



Guardamos y salimos del archive.

**1.2.1. Imagen de Alan Turing**

Ahora que tenemos todo configurado para el segundo sitio, vamos a terminar por subir la imagen al servidor en AWS, para poder mostrarla de forma local.

Para empezar, aquí se puede ver cómo la línea de código en mi HTML que carga la imagen la utiliza localmente, y no hace uso de un enlace externo:



Vamos a hacer uso programa “**[WinSCP](https://winscp.net/eng/index.php)**”[🔗](https://winscp.net/eng/index.php), el cual se presupone que ya está instalado.

Otras opciones/formas de realizar este procedimiento son las siguientes:

- **Comando “[scp](https://linuxize.com/post/how-to-use-scp-command-to-securely-transfer-files/)”**[**🔗**](https://linuxize.com/post/how-to-use-scp-command-to-securely-transfer-files/)**:** Este comando nos permite realizar una conexión con el servidor que queramos para poder transferir archivos de forma segura.

- **IDE:** Podemos utilizar un IDE que nos muestre un entorno gráfico para arrastrar con el ratón el archivo deseado hacia el servidor remoto.

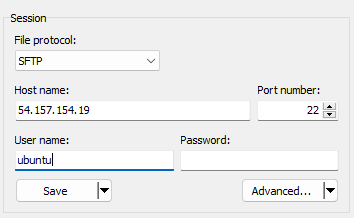
- **Comandos “[curl](https://blog.hubspot.com/website/curl-command" \l ":~:text=API%20%5BFree%20Ebook%5D-,What%20is%20the%20cURL%20command%3F,-Client%20URL%20(cURL)”**[**🔗**](https://blog.hubspot.com/website/curl-command#:~:text=API%20%5BFree%20Ebook%5D-,What%20is%20the%20cURL%20command%3F,-Client%20URL%20(cURL) **o “[wget](https://www.hostinger.com/tutorials/wget-command-examples/)”**[**🔗**](https://www.hostinger.com/tutorials/wget-command-examples/)**:** Distintos comandos que permiten descargar la imagen en lugar de transferirla.

Una vez abrimos el programa “WinSCP” se nos muestra un cuadro donde ingresaremos las credenciales necesarias para realizar la conexión con el servidor, algo parecido a la conexión mediante PuTTy/MobaXterm.

El protocolo lo dejamos en “[**SFTP**](https://www.ssh.com/academy/ssh/sftp-ssh-file-transfer-protocol#:~:text=SFTP%20(SSH%20File%20Transfer%20Protocol,is%20quickly%20replacing%20FTP%2FS.)”[🔗](https://www.ssh.com/academy/ssh/sftp-ssh-file-transfer-protocol#:~:text=SFTP%20(SSH%20File%20Transfer%20Protocol,is%20quickly%20replacing%20FTP%2FS.).

En el recuadro de “**Host name**” ingresamos la dirección IP pública de nuestra instancia de AWS. El puerto lo dejamos en el predeterminado, el 22.

Como usuario utilizaremos el usuario “**ubuntu**” de nuevo.



Por último, entraremos en el botón “**Advanced…**”, donde seleccionaremos la pestaña “**Authentication**” en la barra lateral izquierda.

Nos dirigimos al apartado “**Private key file:**” y seleccionamos la clave vockey con extensión “**.ppk**” que nos descargamos desde el workbench de AWS en prácticas anteriores.

Si la clave no funciona será necesario descargarla de nuevo, ya que puede que haya expirado.



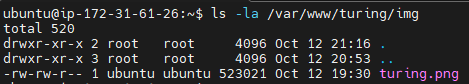
Hacemos clic en “OK” y luego en el botón “**Login**”.

Se nos presenta una interfaz totalmente interactiva donde podremos **arrastrar con el ratón desde la izquierda (máquina local)** los archivos que deseemos llevar a la máquina remota (derecha).

Ubicamos el archivo de la imagen que nos hemos descargado de Alan Turing y la arrastramos hacia la derecha, al que debería ser el directorio que aparece por defecto (“**/home/ubuntu**”). No podemos llevarla directamente al directorio deseado porque necesitamos de unos permisos que no podemos utilizar en WinSCP.

La imagen ya ha sido subida al servidor. Ya podemos cerrar WinSCP.

Volvemos al terminal del servidor, donde usaremos el comando “**sudo mv /home/ubuntu/turing.png /var/www/turing/img/turing.png**” para mover la imagen de sitio hacia el directorio donde apunta nuestro código HTML.



Ahora sólo nos falta ejecutar el siguiente comando para habilitar el sitio que ya hemos configurado y recargar Apache: “**sudo a2ensite turing.conf && sudo systemctl reload apache2**”.



Para terminar con este segundo sitio, volvemos a abrir el archivo “**hosts**” de nuestra máquina local y agregamos el siguiente contenido al final:



Ahora sí, si ingresamos una de estas tres URLs en un navegador web de nuestra máquina local, podremos visitar la página web de Alan Turing.



**1.3. Página no encontrada**

Ya tenemos unas URLs que redireccionan a una página sobre Alan Turing y otras cuantas más que redireccionan a una página de información sobre el alumno.

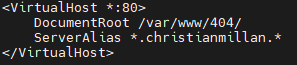
Para realizar el tercer sitio, debemos generar una serie de URLs que actuarán como “fallback” para cuando nos equivoquemos (o entremos de forma premeditada) intentando entrar a la página de información sobre el alumno, de forma que se muestre una página de error personalizado que avise de que no existe y se encuentra en la URL equivocada.

El proceso es muy similar al de los dos sitios anteriores.

Utilizamos primero el comando “**sudo cp /etc/apache2/sites-available/turing.conf /etc/apache2/sites-available/404.conf**”.

Entramos a editar el archivo con “**sudo nano /etc/apache2/sites-available/404.conf**”.

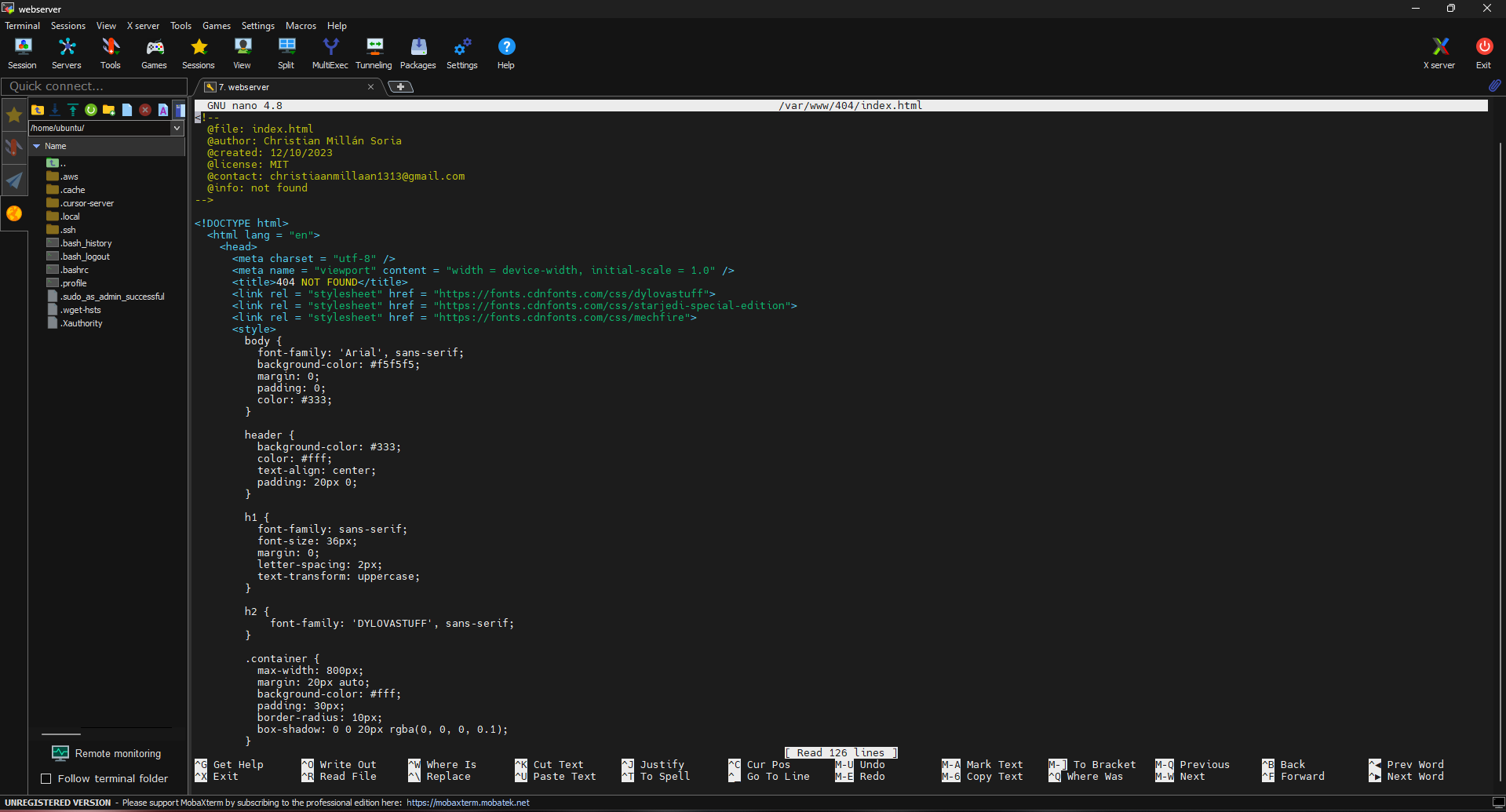
Una vez dentro cambiamos el contenido de forma similar a como lo hicimos con el de Turing, pero escribiendo lo siguiente:



Esta configuración hará que cualquier URL con cualquier terminación de las que aparecen en el alias se redirija aquí, pero si es “**.com**”, “**.net**” o “**.es**” se enviará al primer sitio.

Ahora creamos el archivo HTML para este tercer sitio: “**sudo mkdir /var/www/404 && sudo nano /var/www/404/index.html**”.

Ingresamos el código HTML que queramos que se muestre en la página de error.



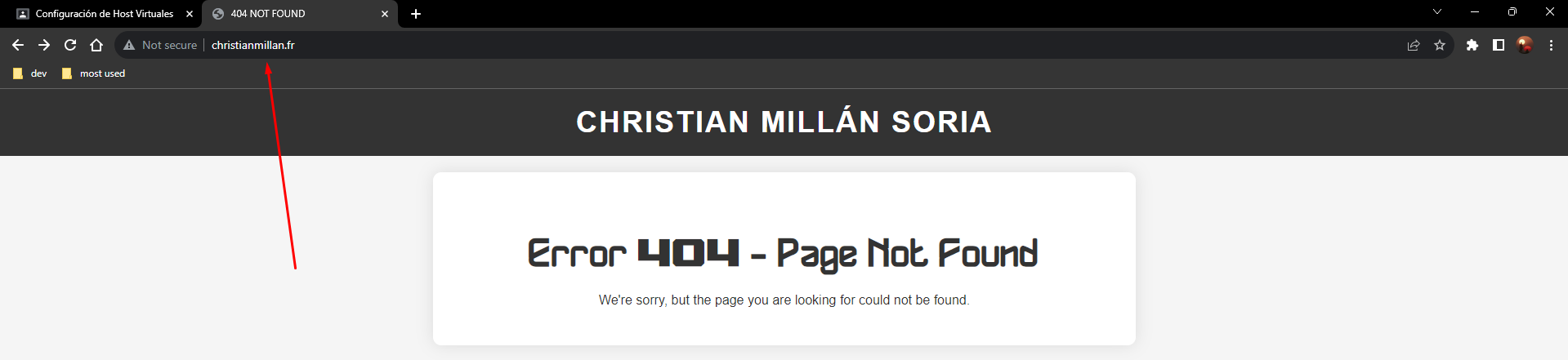
Habilitamos el sitio y recargamos Apache con el comando “**sudo a2ensite 404.conf && sudo systemctl reload apache2**”.



Volvemos a abrir el bloc de notas de la máquina local en modo administrador y escribimos lo siguiente al final:

**{dirección-ip-pública-instancia} \*.{nombre-alumno}.\*** ->(tantas veces como URLs queramos añadir)

Ahora probamos a visitar, por ejemplo, la URL **www.{nombre-alumno}.fr** en el navegador web local.



**2. IP pública elástica**

Como hemos podido comprobar, cada vez que el laboratorio se apaga y lo volvemos a encender la dirección IP pública de nuestro servidor/instancia de AWS cambia, debido a que hay un número limitado de ellas en el mundo y no estamos pagando por ella.

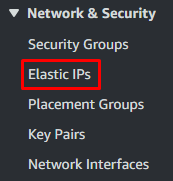
Esto puede ser algo incómodo debido que hay que ir renovándola cada vez que pasa, es por esto que AWS cuenta con una opción llamada “**[Elastic IP Address](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html)**”[🔗](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html).

Con la IP pública elástica podemos ahorrarnos este proceso tedioso.

Para este apartado final de la práctica, vamos a asociar una nueva IP pública elástica a nuestra instancia de AWS.

**2.1. Creación**

Nos dirigimos al [**panel de control de EC2🔗**](https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Home:). En el panel lateral izquierdo, nos dirigimos a la pestaña “**Network & Security**”, donde haremos clic en la opción “**[Elastics IPs](https://us-east-1.console.aws.amazon.com/ec2/home" \l "Addresses:)**”[🔗](https://us-east-1.console.aws.amazon.com/ec2/home#Addresses:).

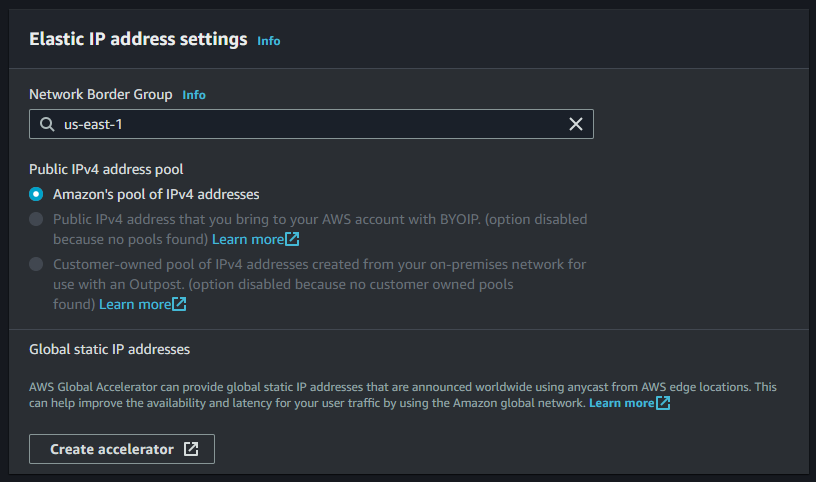


Hacemos clic en el botón “**Allocate Elastic IP address**” de color naranja en la esquina superior derecha.



Se abre una página de configuración para la nueva IP pública elástica.

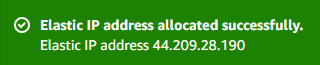
Dejamos la primera opción por defecto.



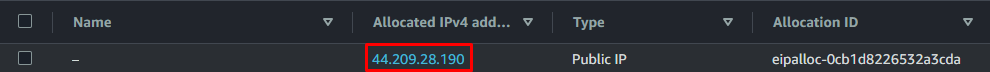
Hacemos clic en el botón “**Allocate**” en naranja abajo a la derecha.



Se muestra un mensaje de confirmación de la creación de la misma.



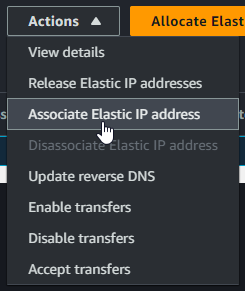
En la lista que nos aparece podemos ver la nueva dirección IP que podremos utilizar.



**2.2. Asociación**

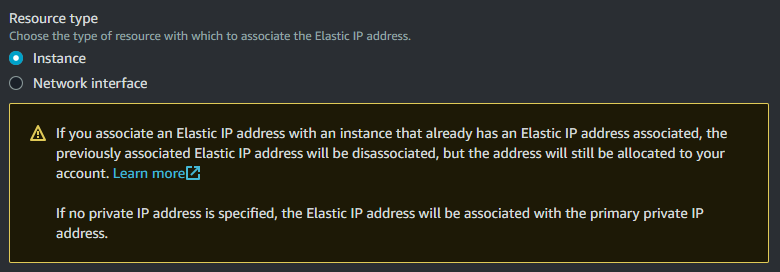
El siguiente paso es asociar esta dirección IP con la instancia deseada, en nuestro caso la instancia “**WebServer**”.

Marcamos la dirección IP en la lista. Hacemos clic en el botón “**Actions**” en la esquina superior derecha. Dentro elegimos la opción “**Associate Elastic IP address**”.

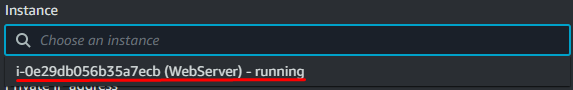


Se muestra un menú para gestionar la asociación.

En el primer apartado “**Resource type**” elegimos la opción “**Instance**”, que es el servicio al que vamos a asociar la IP.



En el apartado “**Instance**”, hacemos clic y nos aparece un desplegable, donde seleccionaremos la instancia correspondiente.



Hacemos clic en el botón “**Associate**”, abajo a la derecha.

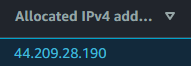


Se muestra de nuevo un mensaje de confirmación.



Si comprobamos el valor de la dirección IP pública elástica que hemos creado en la lista de direcciones podemos ver que coincide con la dirección IP pública que ahora aparece en el resumen de la instancia a la que la hemos asociado en el menú de instancias de EC2.

**Dirección IP pública elástica:**



**Dirección IP pública de la instancia:**



Además, podemos ver un nuevo apartado a la derecha del resumen que ha aparecido después de este proceso:

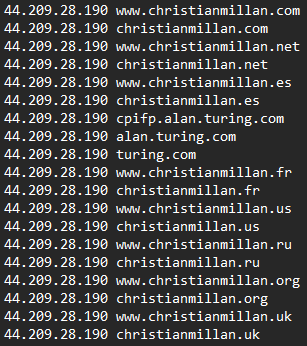


Si hacemos la prueba de apagar el laboratorio de AWS y lo volvemos a encender, veremos que la IP pública de la instancia ahora no cambia, lo que hace que las conexiones y demás procesos no sean tan repetitivos y sean ahora más directos y rápidos.

**2.3. Testeo**

Si hacemos el cambio de IP en el archivo de hosts de nuestra máquina local y volvemos a introducir la URL (cualquiera de las configuradas en esta práctica) podemos comprobar cómo el cambio de IP a una elástica no ha afectado al servidor o provocado un mal funcionamiento del mismo.

**Archivo de hosts:**



**Introducción de una de las URLs:**



