Práctica-Servidor Web.

Hecho por: David Tardío Montoya, 2 GM B, Servicios en Red.

Índice

[**1: Explicación del enunciado. 1**](#_Toc94613204)

[**2: Preparación general y configuración del cliente. 1**](#_Toc94613205)

[**3: Preparación y configuración de red del servidor. 2**](#_Toc94613206)

[**4: Configuración de la primera página web. 4**](#_Toc94613207)

[**5: Configuración de la segunda página web. 7**](#_Toc94613208)

[**5: Configuración de la última página web. 9**](#_Toc94613209)

# 1: Explicación del enunciado.

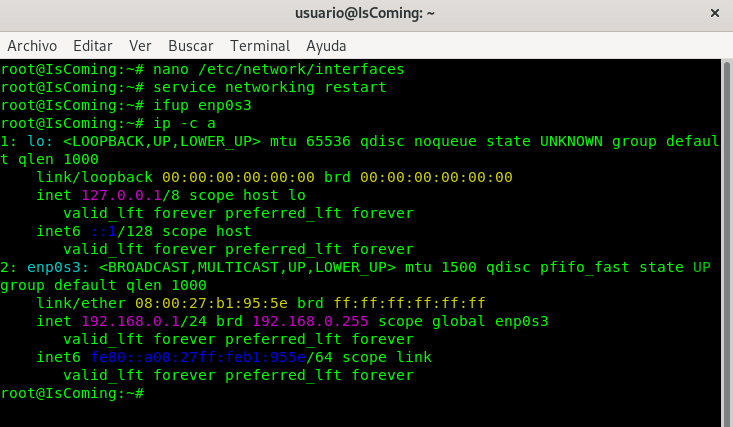
Hoy, vamos a crear un servidor web Apache en Debian sin entorno. Este servidor tendrá una única interfaz de red conectada por red interna al cliente, aun teniendo dos IPs en esa interfaz. Deberá tener tres sitios web, con diversas características, accesibles de diversas formas.

# 2: Preparación general y configuración del cliente.

Ahora, deberemos crear dos máquinas Debian. Una que sea exclusivamente terminal (este será el **servidor**), y otra que puede o no tener interfaz (ya que se tratará del **cliente**). En mi caso, el cliente tendrá una interfaz gráfica para diferenciarlo con más facilidad.

Tenemos que asegurar que el cliente (el servidor todavía debe esperar) esté en una **red interna**, a la cual bautizaremos con el nombre que queramos (**Yon Wick**¸ por ejemplo). Tras ello, arrancaremos la máquina cliente. Pondremos diversas IP’s al servidor, que comenzarán desde la 192.168.0.80 (por pura comodidad), por lo que haremos que **la IP del cliente sea la 192.168.0.1.**

Para esto, iremos a su archivo de configuración de interfaces de red, usando el comando ***nano /etc/network/interfaces***. Haremos que la IP sea estática, y que tenga esa IP, junto a otras configuraciones (como la puerta de enlace o la máscara de red). Reiniciamos el servicio usando el comando ***service networking restart***, y levantaremos el adaptador de red usando el comando ***ifup enp0s3***, y comprobaremos que haya recibido la IP correcta usando el comando ***ip –c a***.



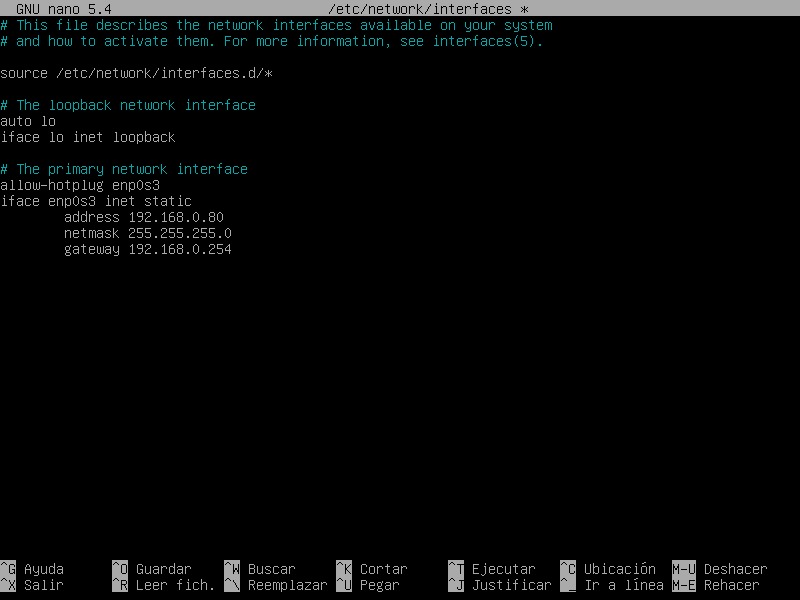
IP del cliente

# 3: Preparación y configuración de red del servidor.

En este paso, realizaremos dos cosas: **asegurar la conectividad entre cliente y servidor; e instalar los paquetes necesarios para configurar el servicio web,** aunque no en ese orden.

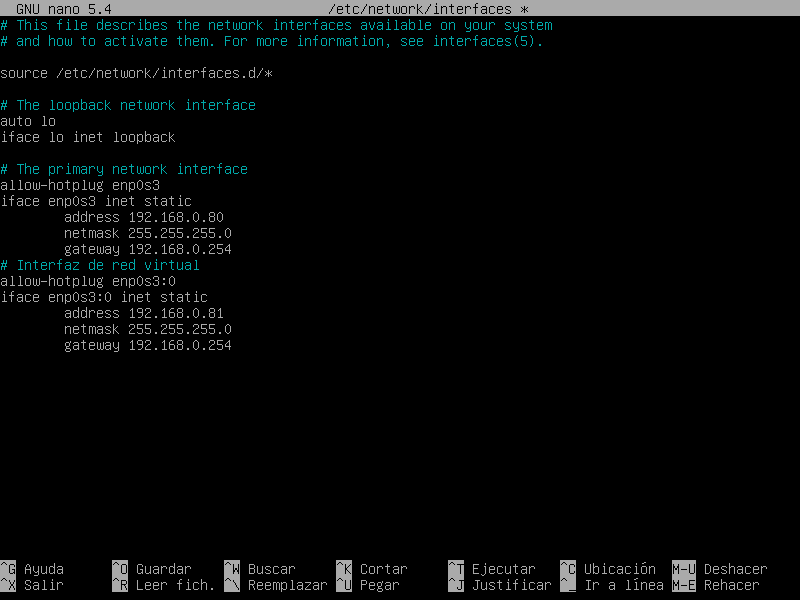
Mientras la máquina este en red NAT, **arrancaremos la máquina servidor**. Una vez ahí, **instalaremos apache**, usando el comando ***apt install apache2 -y***. Puede ser necesario hacer un *apt update* y un *apt upgrade*. Una vez Apache esté instalado, iremos al archivo de configuración de los adaptadores de red (que es el mismo archivo que editamos en el cliente), usando el comando ***nano /etc/network/interfaces***.

Sin embargo, **debemos tener más de una dirección IP en una única interfaz**, por lo que esta vez será ligeramente distinto. Empezaremos poniendo la primera dirección IP, de la misma forma que hicimos anteriormente, pero poniendo la **192.168.0.80** como primera IP.



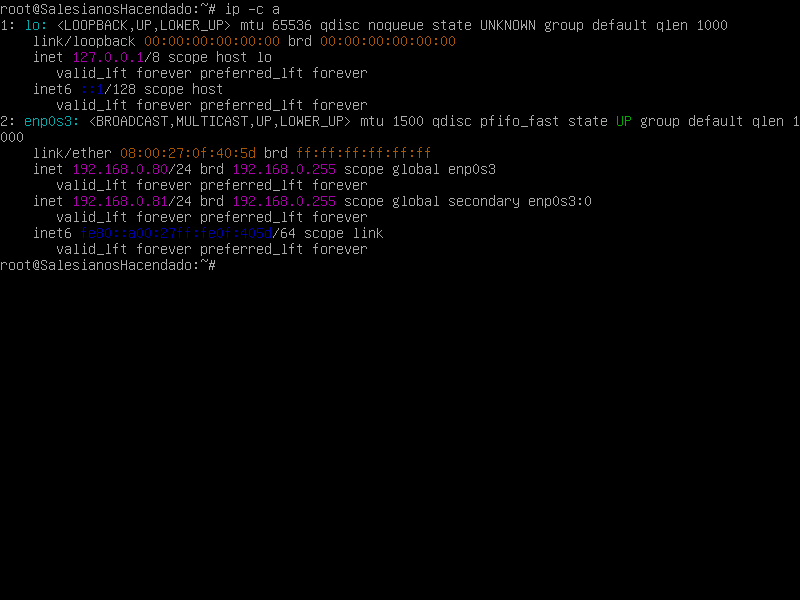
Primera dirección IP

Ahora, debajo de ello tendremos que poner **allow-hotplug enp0s3:0**. Esto creará una **interfaz de red virtual**, a la que podremos configurar una IP de la misma forma que una interfaz normal. A esta interfaz virtual pondremos la **192.168.0.81**.



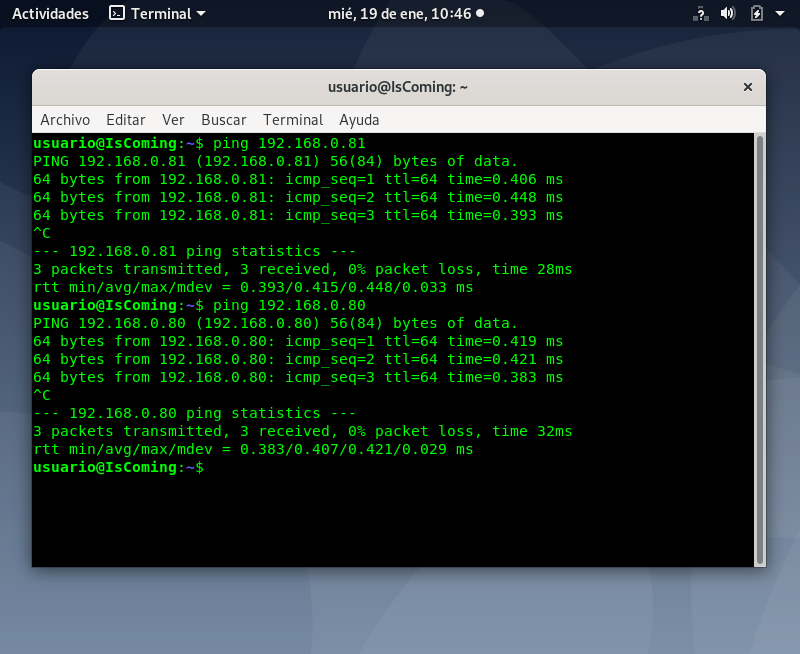
Segunda dirección IP configurada

Ahora, resta usar un ***service networking restart***, un ***ifup enp0s3****, y un* ***ifup enp0s3:0****.* Nos aseguramos de que estamos en red interna, y lo comprobamos con ***ip –c a***.



IPs configuradas

Haremos una prueba de conectividad desde **Cliente**, usando ***ping 192.168.0.80***, y ***ping 192.168.0.81***.



Conectividad comprobada

# 4: Configuración de la primera página web.

Tenemos varios requisitos de esta primera página web:

-Que sea accesible desde la **192.168.0.80**, en el puerto **80**.

-Que se aloje en el sitio **/var/www/html1.**

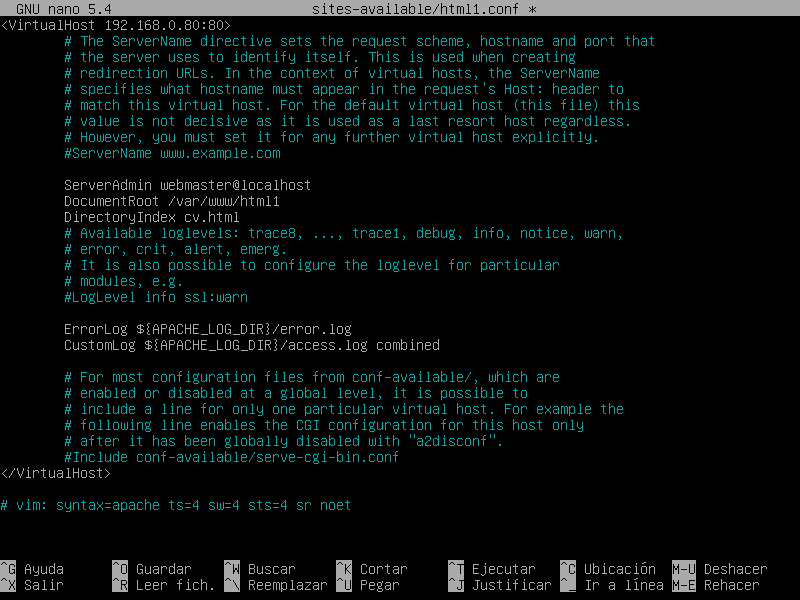
-Un HTML llamado **cv.html**, con un documento hecho en html.

-Que no exista ningún archivo llamado index.html en este sitio.

-Que se cargue el currículo directamente sin ponerlo en la URL.

-También se debe crear un archivo .htaccess que impida listar el contenido desde el navegador.

Aun siendo muchas cosas, iremos por partes. Primero, crearemos nuestro sitio. Para ello, nos moveremos al directorio de los sitios disponibles. Usaremos el comando ***cd /etc/apache2/sites-available/***. Haremos un ls, y copiaremos el archivo por defecto usando el comando ***cp 000-default.conf html1.conf.*** Una vez hecho, editaremos el sitio usando ***nano html1.conf.*** Una vez dentro, iremos a la opción **DocumentRoot** y editaremos el fichero para que apunte a **/var/www/html1**. También, en el inicio de la línea, deberemos poner **<VirtualHost 192.168.0.80:80>**. Por último, también añadiremos la directiva **DirectoryIndex cv.html**, para que busque un archivo con ese nombre y no el Index.html.



Configuración del archivo

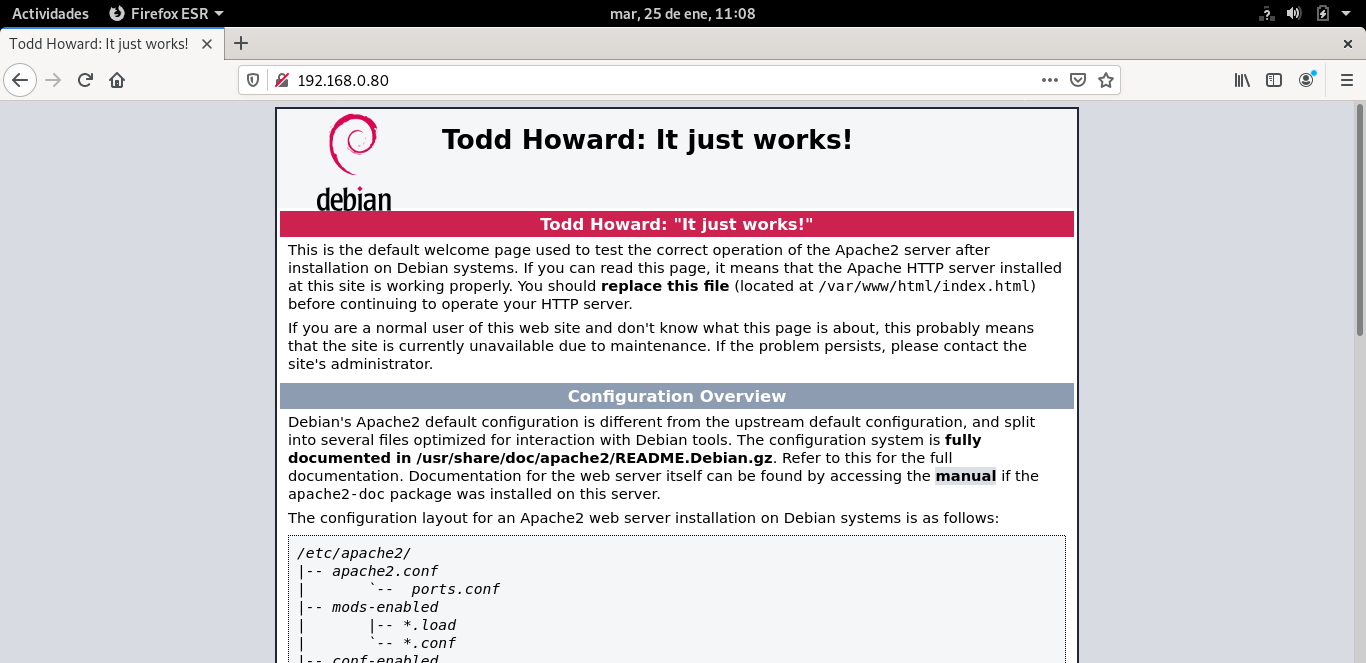
Activaremos el sitio usando el comando ***a2ensite html1.conf.*** No será necesario editar el archivo **ports.conf**, ya que, por defecto, el programa escucha por el puerto 80. Pedirá recargar apache2, pero primero seguiremos configurando nuestro sitio.

Iremos al archivo donde está el index.html por defecto, usando ***cd /var/www***. Una vez ahí, copiaremos el directorio por defecto (html) y lo pegaremos con el nombre que hemos indicado anteriormente (html1). Para ello, usaremos el comando ***cp –R html html1***. Ahora, tendremos que cambiar el nombre del Index.html por cv.html para hacer comprobaciones. Lo haremos usando el comando ***mv Index.html cv.html.***

******

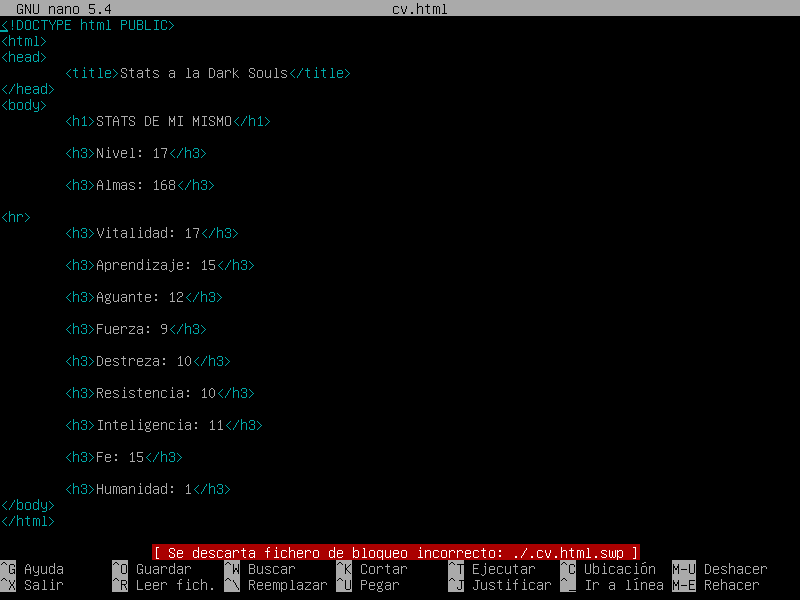
Página con nombre cambiado

Haremos una pequeña diferencia al cv.html (aunque sea básicamente una copia del index.html), editándolo usando ***nano cv.html.*** Se hará una diferencia poco notable. Usaremos el comando ***service apache2 reload*** para reiniciarlo y hacer pruebas. Una vez reiniciado, iremos desde el cliente, y en un buscador pondremos ***192.168.0.80:80***.



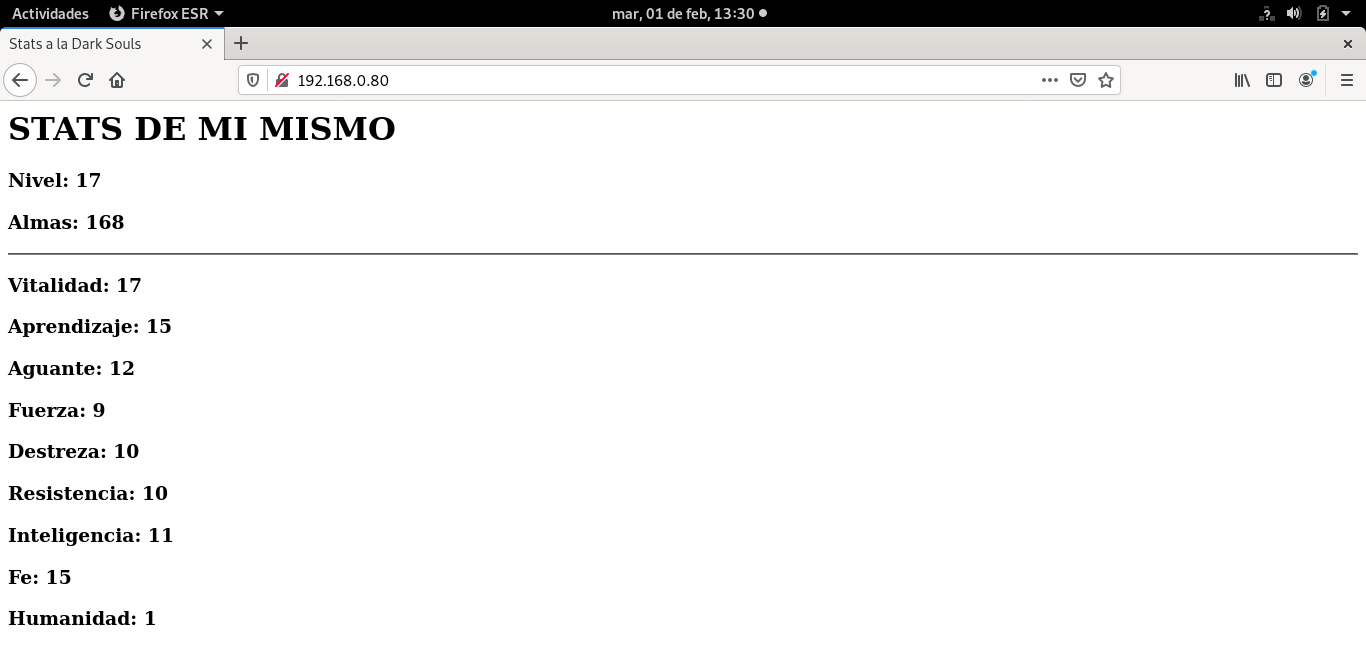
It just Works!

Tras esto, editaremos con ***nano cv.html*** para darle la forma que queramos. Será algo muy sencillo, y rudimentario.



Edición del archivo

Lo comprobaremos desde el cliente.



Página realizada

# 5: Configuración de la segunda página web.

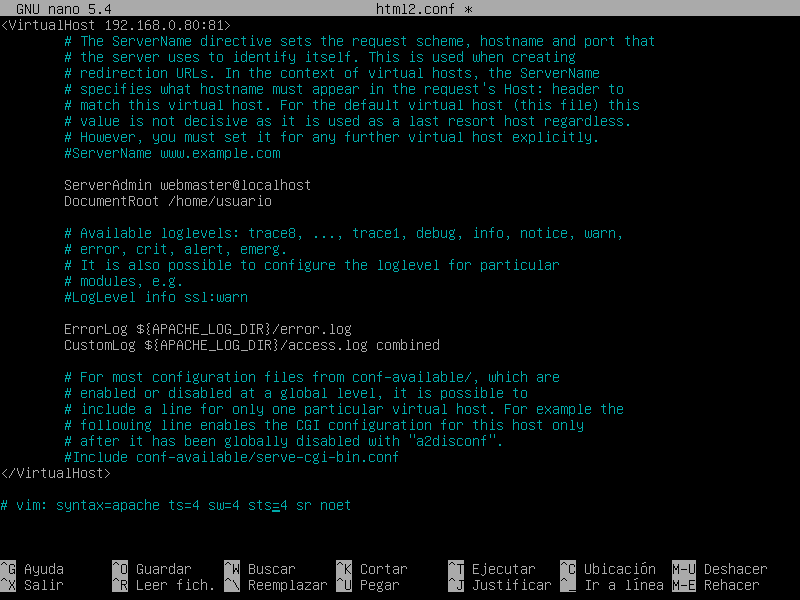
Los requisitos de esta segunda página web son:

-El segundo sitio debe estar en el directorio **/home/usuario**.

-Debe tener un Index.html, además de sustituir levemente este archivo para que se note.

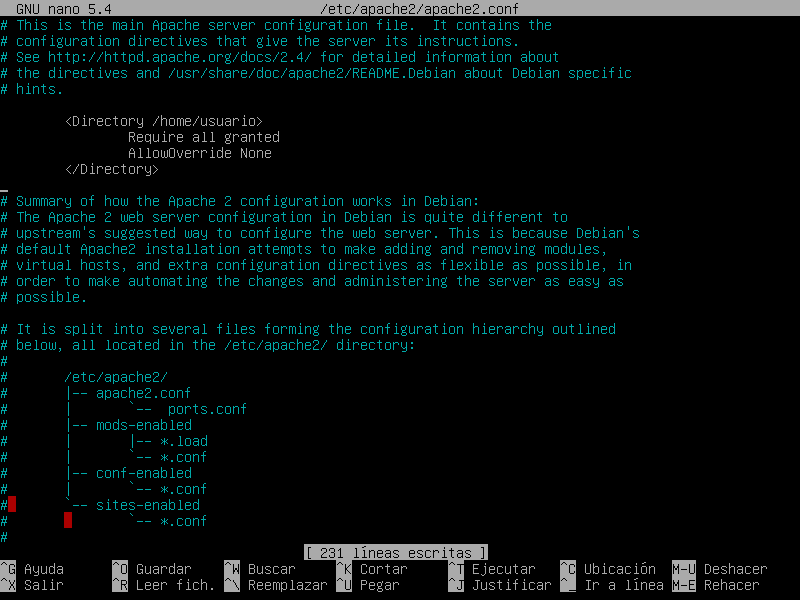
-Debe ser accesible desde la 192.168.0.80 en el puerto 81.

Empezaremos creando nuestro sitio, yendo al archivo de los sitios disponibles. Usaremos el comando ***cd /etc/apache2/sites-available/***. Una vez ahí, copiaremos la plantilla de página web usando el comando ***cp 000-default.conf html2.conf.*** Tendremos que editar el sitio usando ***nano html2.conf***. Una vez aquí, cambiaremos el VirtualHost, y pondremos **192.168.0.80:81.** Cambiaremos el DocumentRoot para poner **/home/usuario**. Esto hará que acepte peticiones por el puerto 81, y que busque el index.html en /home/usuario.



Configuración de la página web realizada.

Por último, añadiremos unas líneas de “Directory” en **/etc/apache2/apache2.conf** que servirá para que se pueda ver el recurso por un cliente, ya que nos dará un error de lo contrario[[1]](#footnote-1).



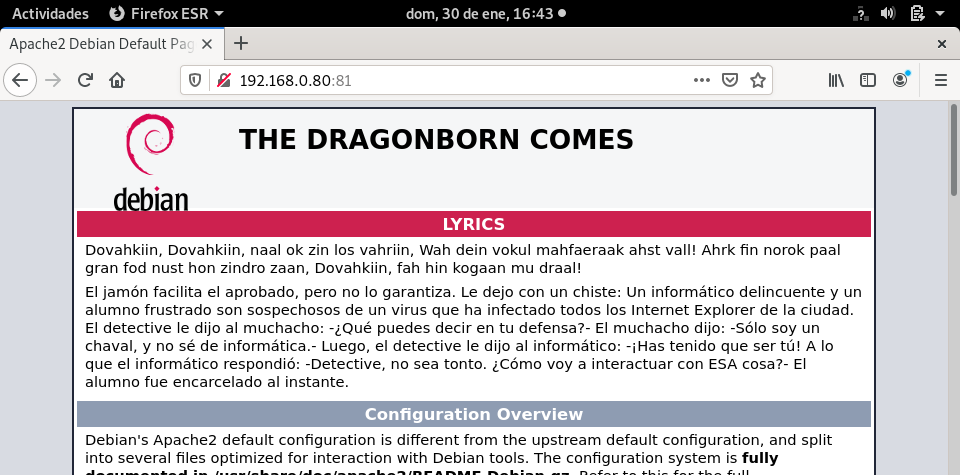
Configuración del fichero.

Ya que esta página se debe acceder por el puerto 81, deberemos editar el archivo de configuración, usando el comando ***nano ports.conf.*** Añadiremos en una línea libre cerca del principio **“Listen 81”**, para que escuche por ese puerto. Ahora, podremos activar el sitio usando ***a2ensite html2.conf***. Reiniciaremos el sitio web después, una vez hayamos movido el index.html a su sitio correspondiente**.** Para ello, usaremos el comando ***cd /var/www/html.*** Una vez dentro, copiaremos el index.html a /home/usuario, usando el comando ***cp index.html /home/usuario/index.html.***



Copia realizada

Ahora, usando ***nano /home/usuario/index.html***, haremos otra edición del archivo. Esta será leve, pero también definitiva. Usaremos el comando ***service apache2 restart*** una vez hecho.



Prueba realizada.

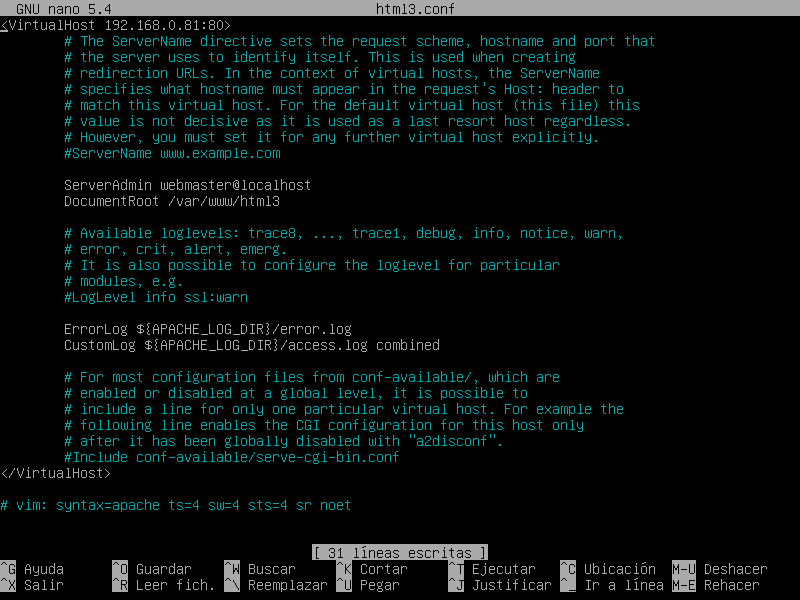
# 5: Configuración de la última página web.

Los requisitos de esta tercera página web son:

-La carpeta se alojará en **/var/www/html3**.

-Debe protegerse con una contraseña el directorio root del sitio, para que sólo un usuario llamado **img** pueda acceder.

Crearemos nuestro sitio, usando el comando ***cd /etc/apache2/sites-available/***. Copiamos la plantilla de página web usando el comando ***cp 000-default.conf html3.conf.*** Editaremos el sitio usando ***nano html3.conf***. Cambiaremos la IP y el puerto al **192.168.0.81:80.** Cambiaremos el DocumentRoot para poner **/var/www/html3**, que servirá para que la carpeta de conexión se realice por esta IP, y que esté en este directorio.

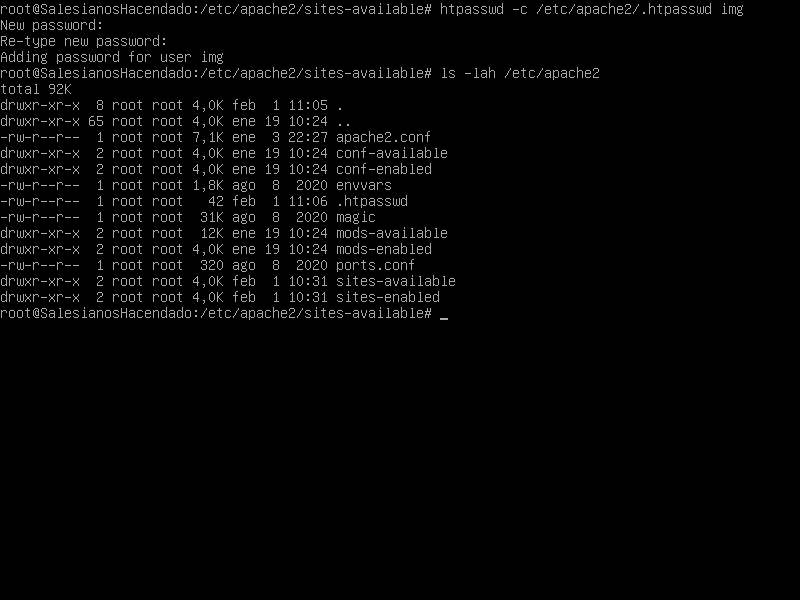


Archivo configurado

Activaremos el sitio usando el comando ***a2ensite html3.conf.*** No será necesario editar el archivo **ports.conf,** pues usaremos el puerto por defecto. Usaremos ***service apache2 restart***.

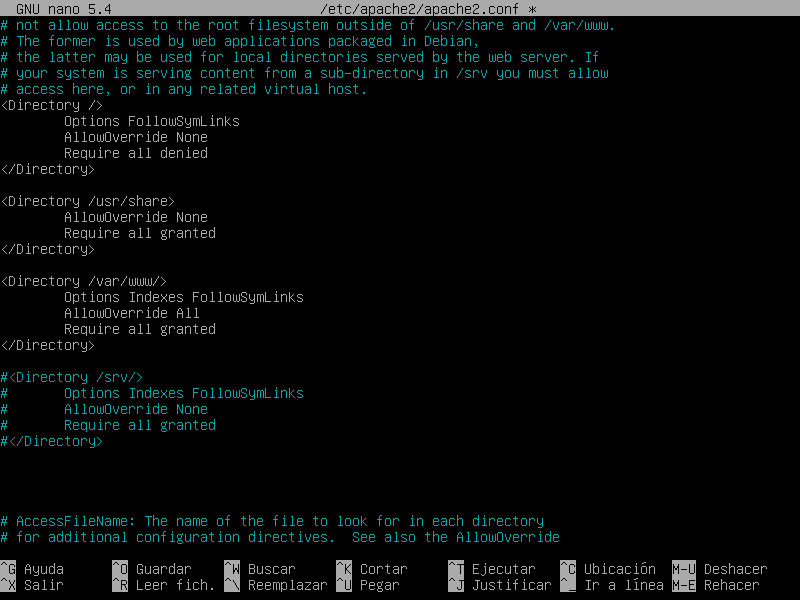
Tras esto, copiaremos el Index.html en html3. Haremos ***mkdir /var/www/html3/*** para crear el directorio de conexión en el que se alojará el Index***.*** Usaremos el comando **cp */var/www/html/index.html /var/www/html3/index.html*** para copiar el Index.

Ahora, tendremos que habilitar la restricción de directorio. Crearemos un archivo que se usará para verificar el usuario y contraseña. Lo haremos usando un comando distinto que sirve para agregarle contraseña. Este comando es ***htpasswd –c /etc/apache2/.htpasswd img***. El **.htpasswd** es el nombre del archivo que crearemos, e **img** será el usuario que podrá acceder. La contraseña será **1234**.



Archivo realizado

Ahora, deberemos configurar el archivo para que Apache permita archivos .htaccess. Para ello, usaremos ***nano /etc/apache2/apache2.conf***, y en la línea **<Directory>**, en /var/www, cambiar el **AllowOverride** a **All.**

****

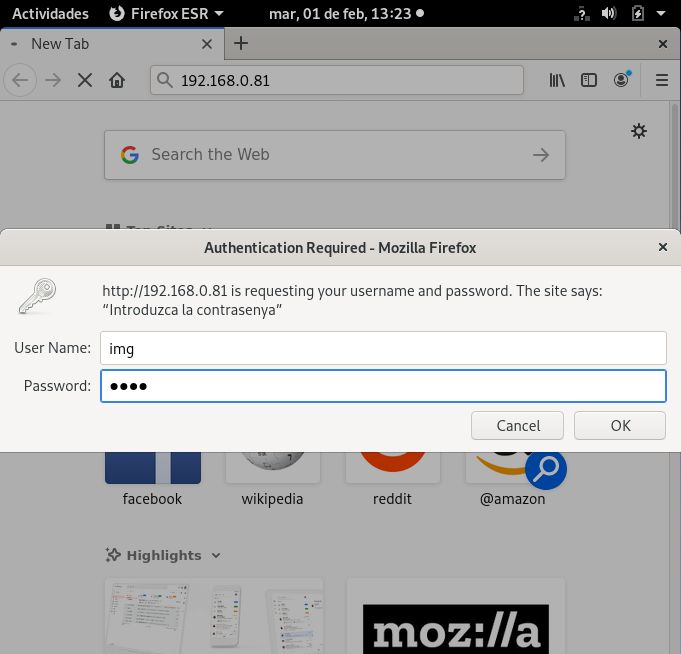
Realizado

Reiniciamos el servicio. Iremos al directorio **/var/www/html3**, y crearemos un archivo llamado **.htaccess**. Escribiremos lo que aparece en la imagen a continuación.



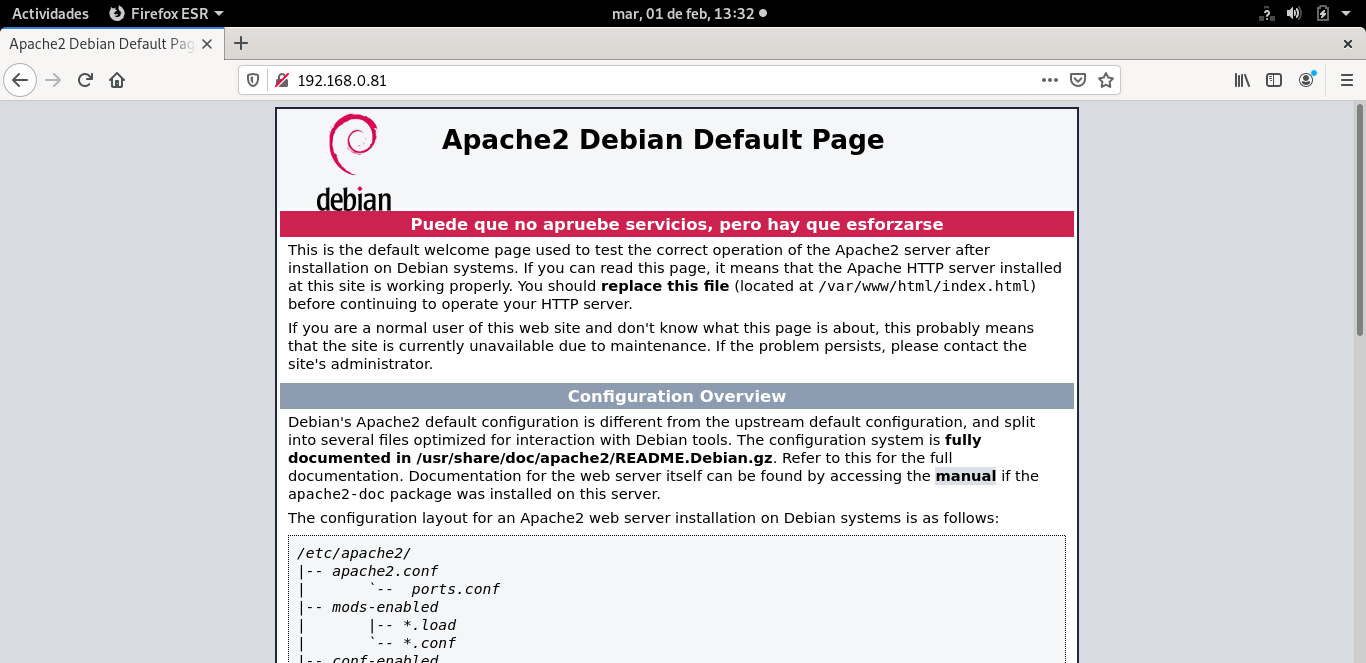
Configuración del archivo

Esto sirve para decirle a Apache que se necesite una contraseña y un usuario válido para acceder a cualquier contenido dentro del directorio. Desde el cliente, pondremos en el buscador ***192.168.0.81:80***, y una vez ahí, nos saldrá la autenticación.



Autenticación

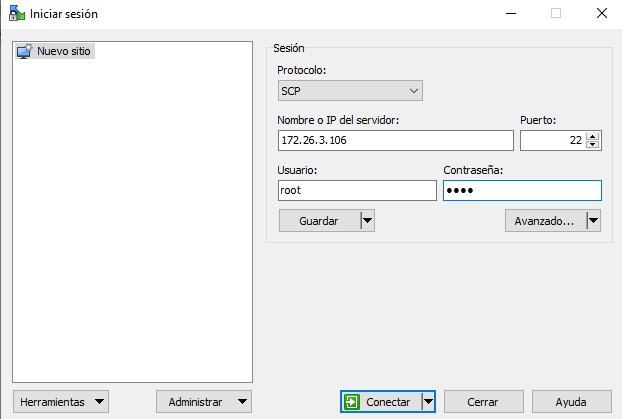
Introducimos las credenciales, y podremos ver una versión que modifiqué ligeramente del Index.html.



Página realizada

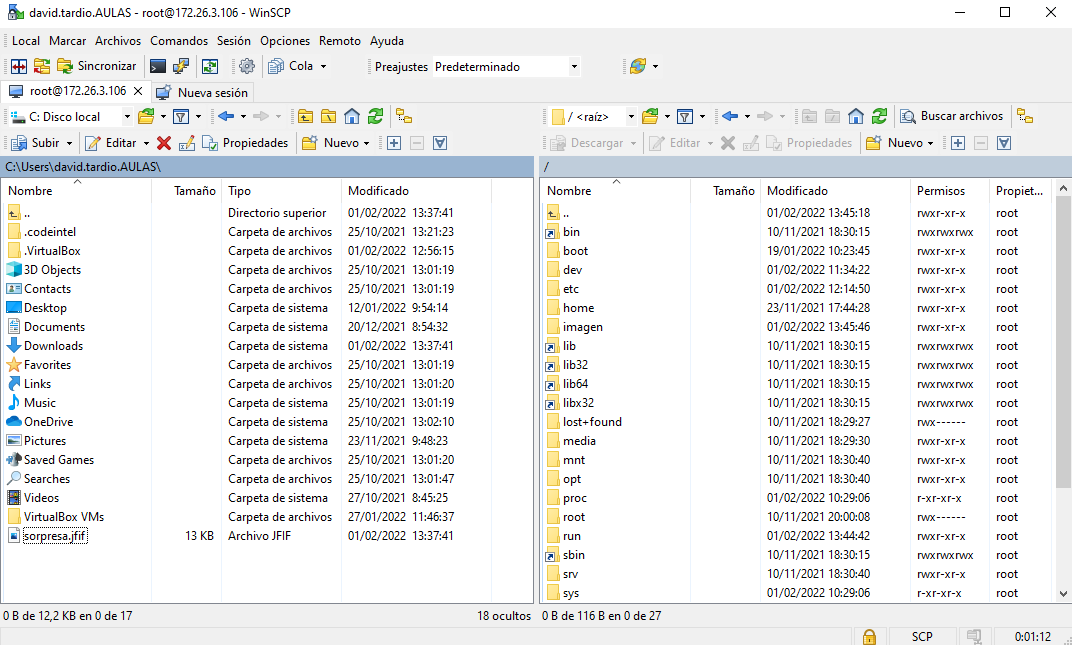
Ahora, para finalizar, un requisito es que el Index contenga una imagen con algo extra. Para esto, tendremos que instalar SSH usando ***apt install ssh*** en el SERVIDOR, tras haber puesto la máquina en **modo puente**, y tras haber puesto que reciba IP de forma dinámica. Como esta máquina sólo tiene a root como (super)usuario, tendremos que configurar SSH para que podamos loguearnos como root. Este paso se obviará, pues no está estrechamente relacionado.

Una vez realizado, en nuestra máquina real, descargaremos una imagen que queramos. Tras hacerlo, en nuestra máquina real, usaremos **WinSCP.** Introduciremos la IP del servidor, el usuario y la contraseña.



Proceso

Seleccionamos “conectar”.



Transacción realizada

Ahora, comprobaremos si está en su directorio correspondiente, en **/imagen**.



La imagen se ha transferido

Volvemos a configurar la red interna. Movemos la imagen “sorpresa” a “/var/www/html3”. Editaremos el Index.html con un poco de HTML básico. Haremos la comprobación final desde el cliente.



Funciona correctamente.

# 6: Bibliografía.

[***https://eltallerdelbit.com/servidor-web-apache/#habilitar\_sitio\_web\_apache***](https://eltallerdelbit.com/servidor-web-apache/#habilitar_sitio_web_apache)

[***https://www.tecmint.com/forbidden-you-dont-have-permission-to-access-on-this-server-error/***](https://www.tecmint.com/forbidden-you-dont-have-permission-to-access-on-this-server-error/)

[***https://www.linuxenespañol.com/tutoriales/proteger-un-directorio-apache-con-contrasena/***](https://www.linuxenespañol.com/tutoriales/proteger-un-directorio-apache-con-contrasena/)

1. Puede ser necesario cambiar los permisos de esa carpeta. [↑](#footnote-ref-1)