|  |
| --- |
|  |
| INSTALACIÓN Y CONFIGURACIÓN DEL SERVIDOR WEB APACHE |
|  |

SERVICIOS EN RED

JUAN CARLOS NAVIDAD GARCÍA

**Índice:**

[1. ¿Qué es HTTP y HTTPS?: 2](#_Toc94814877)

[2. ¿Qué es Webmin?: 3](#_Toc94814878)

[3. ¿Cómo instalar Webmin?: 3](#_Toc94814879)

[4. ¿Qué es Apache?: 5](#_Toc94814880)

[5. ¿Cómo instalar Apache?: 6](#_Toc94814881)

[6. Funcionamiento de Apache en Webmin: 6](#_Toc94814882)

[7. ¿Qué son los módulos Apache?: 10](#_Toc94814883)

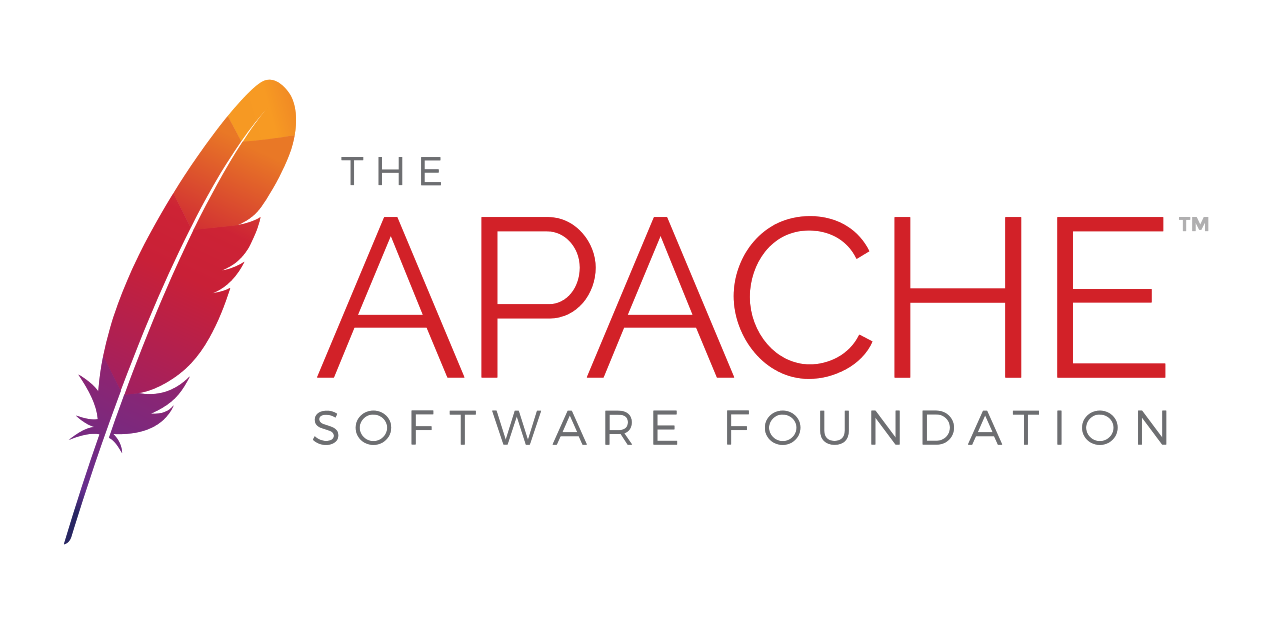
[8. ¿Cómo funciona y se configura el módulo UserDir?: 10](#_Toc94814884)

[9. Comprobación de funcionamiento: 14](#_Toc94814885)

[10. ¿Qué son los ficheros .htaccess?: 15](#_Toc94814886)

[11. ¿Cómo crear ficheros .htaccess?: 16](#_Toc94814887)

[12. Bibliografía: 18](#_Toc94814888)



### ¿Qué es HTTP y HTTPS?:

HTTP, de sus siglas en inglés: "Hypertext Transfer Protocol", es el nombre de un protocolo el cual nos permite realizar una petición de datos y recursos, como pueden ser documentos HTML. Es la base de cualquier intercambio de datos en la Web, y un protocolo de estructura cliente-servidor.

Básicamente se basa en sencillas operaciones de solicitud/respuesta. Un cliente establece una conexión con un servidor y envía un mensaje con los datos de la solicitud. El servidor responde con un mensaje similar, que contiene el estado de la operación y su posible resultado.

En cuanto al servidor web o servidor HTTP, es un software que forma parte del servidor y tiene como misión principal devolver información (páginas) cuando recibe peticiones por parte de los usuarios.

En otras palabras, es el software que permite que los usuarios que quieren ver una página web en su navegador puedan hacerlo.

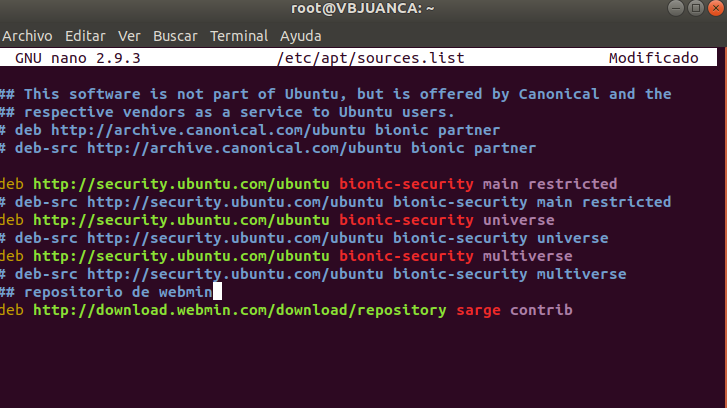
Por otra parte, HTTPS (HyperText Transfer Protocol Secure, protocolo seguro de transferencia de hipertexto), es lo mismo pero la comunicación con el cliente está totalmente cifrada. Lo que hace este protocolo aún más seguro.

### ¿Qué es Webmin?:

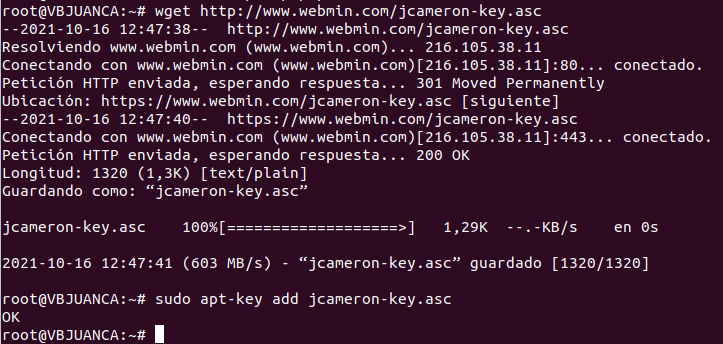
Para realizar esta práctica, vamos a utilizar un sistema Linux con Apache2 mediante la interfaz gráfica de Webmin. Como ya he dicho en prácticas anteriores Webmin es una herramienta que nos proporciona una interfaz gráfica intuitiva y fácil de usar para administrar el sistema, servidores, etc.

### ¿Cómo instalar Webmin?:

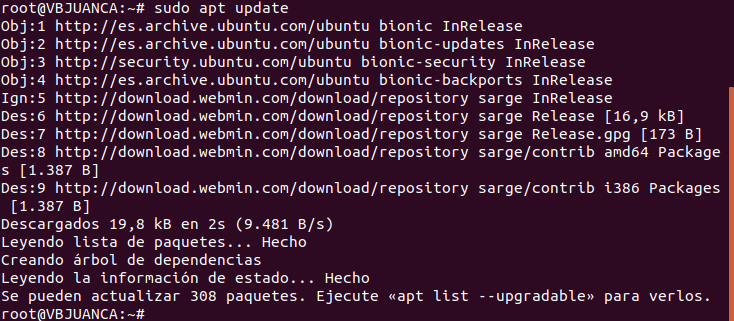
* En primer lugar, debemos añadir el repositorio **Webmin** para poder instalar y actualizar **Webmin** fácilmente usando nuestro **administrador de paquetes** **/etc/apt/sources.list**. Esto se hace agregando el repositorio:
* **deb http://download.webmin.com/download/repository sarge contrib**



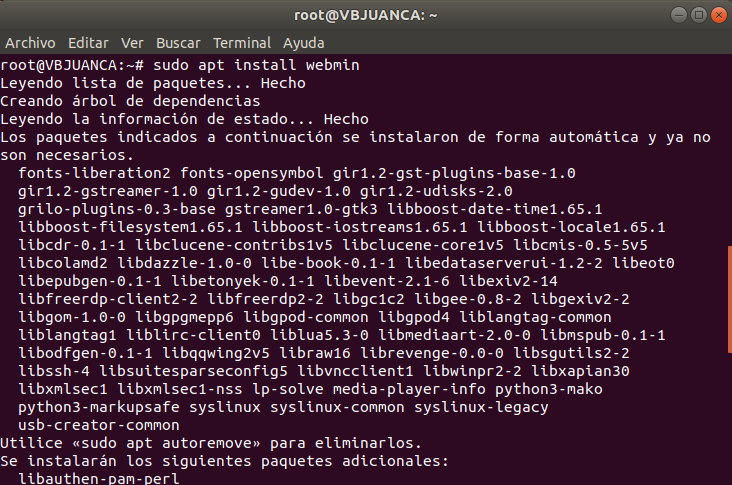
* A continuación, agrego la **clave PGP de Webmin** para que el sistema confíe en el nuevo repositorio:
* **wget http://www.webmin.com/jcameron-key.asc**
* **sudo apt-key add jcameron-key.asc**



* Luego, actualizo la lista de paquetes para que incluya el repositorio **Webmin**:
* **sudo apt update**



* Finalmente instalo **Webmin**:
* **sudo apt install webmin**



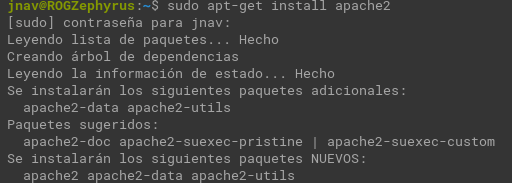
### ¿Qué es Apache?:

El Servidor Web Apache es un software de servidor web gratuito y de código abierto multiplataforma con el cual se ejecutan el 46% de los sitios web de todo el mundo. Es mantenido y desarrollado por la Apache Software Foundation.

Les permite a los propietarios de sitios web servir contenido en la web, de ahí el nombre de <servidor web>. Es uno de los servidores web más antiguos y confiables, con la primera versión lanzada hace más de 20 años, en 1995.

Cuando alguien quiere visitar un sitio web, ingresa un nombre de dominio en la barra de direcciones de su navegador. Luego, el servidor web envía los archivos solicitados actuando como un repartidor virtual.

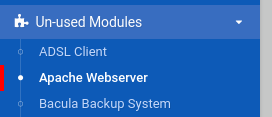
### ¿Cómo instalar Apache?:

Instalar Apache es muy simple, así que cuando tengamos instalado Webmin, a no ser que ya lo tengamos instalado, introduciremos el siguiente comando en una terminal para poder instalar el paquete Apache:

* **sudo apt-get install apache2**

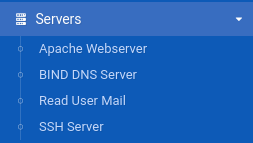
### Funcionamiento de Apache en Webmin:

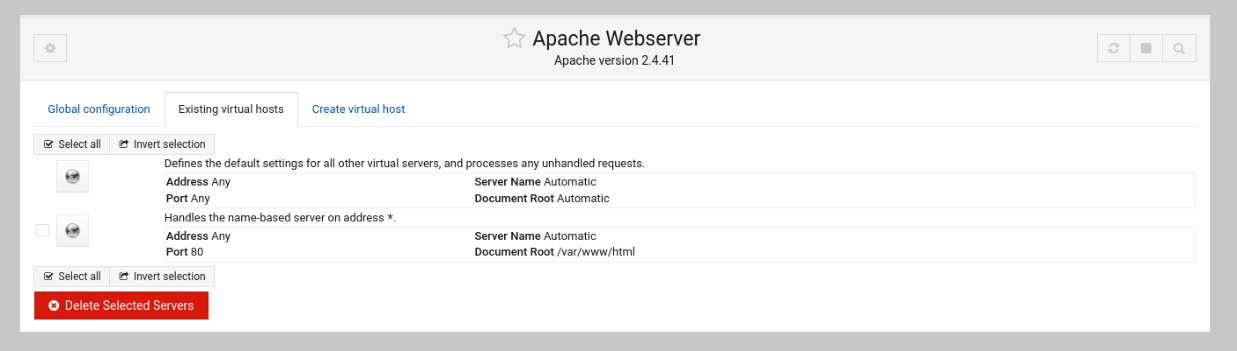
Una vez tengamos instalado el servidor web Apache, nos iremos a Webmin, recordando que se accede desde el navegador escribiendo localhost:10000 o <ip>:10000.



Una vez dentro de Webmin, en el panel de contenido de la parte izquierda, nos iremos al apartado *Un-used Modules*, donde nos encontraremos con Apache Webserver.

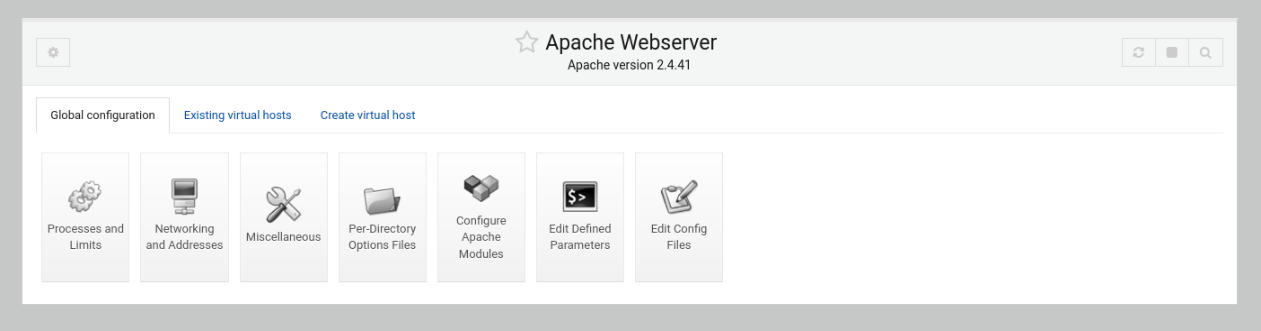
Podemos hacer que el servidor Apache, suba al apartado de servidores clicando en el botón *Refresh Modules*.

Después de pulsar el botón *Refresh Modules*, Webmin se recargará y ahora, nos saldrá Apache Webserver en el apartado Servers con todos nuestros demás servidores.

Si accedemos a Apache Webserver, la pantalla principal será la de Hosts virtuales existentes:

Esta pantalla corresponde a la de los Hosts virtuales activos o existentes, Un host virtual, o VirtualHost, en Apache nos permite mantener múltiples nombres de host en nuestro servidor y ubicar en cada uno de ellos una o varias páginas HTML.

La primera pantalla es la de Configuración local:

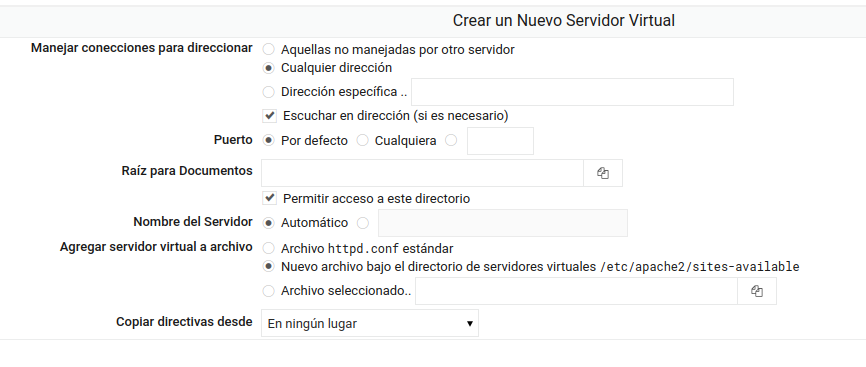


Podemos configurar aquí:

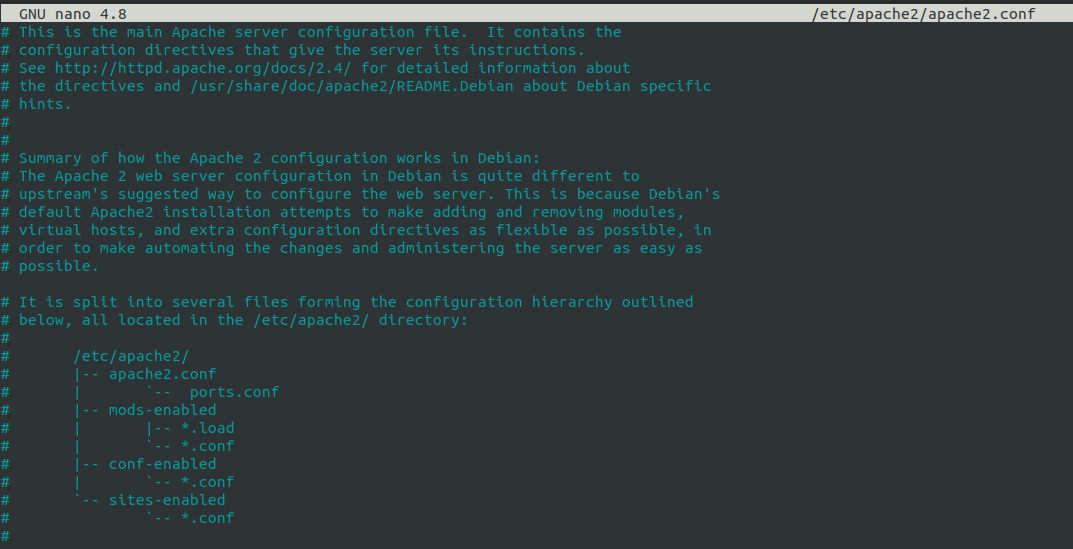
* Límites y procesos.
* Redes y Direcciones.
* Tipos MIME
* Creación de ficheros .htaccess
* Configuración de módulos.
* Edición de parámetros.
* Edición de los ficheros de configuración.

La última pantalla correspondería a la creación de hosts virtuales.

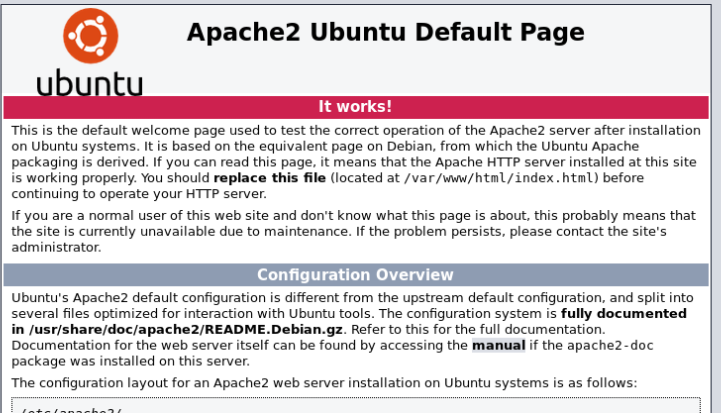
En esta pantalla podremos crear y medio configurar un nuevo host virtual, lo cual ya he explicado anteriormente. Tendrá todos los parámetros a configurar que se pueden observar en la captura.



A parte de Webmin, como se hace en otros servidores, también se pueden configurar manualmente, el archivo de configuración principal de apache es */etc/apache2/apache2.conf* y tendría la siguiente forma:



Por otro lado, si lo que queremos es configurar la página web principal de Apache, nos debemos de ir al directorio */var/www/*, allí se encontrará el index.html, la página principal de Apache. Esta la podremos modificar por cualquier otra, incluso hacer uso de javascript, hojas de estilos, básicamente páginas web completas.

La página principal de Apache sería así:

En mi caso, he modificado el código del */var/www/index.html* por este:



La página se vería así:

### ¿Qué son los módulos Apache?:

Un módulo es una parte independiente de un programa. La mayor parte de la funcionalidad de Apache está contenida en módulos que pueden incluirse o excluirse.

Entonces, el servidor HTTP Apache es un programa modular en el que el administrador puede elegir qué funcionalidades se incluyen mediante la selección de un conjunto de módulos.

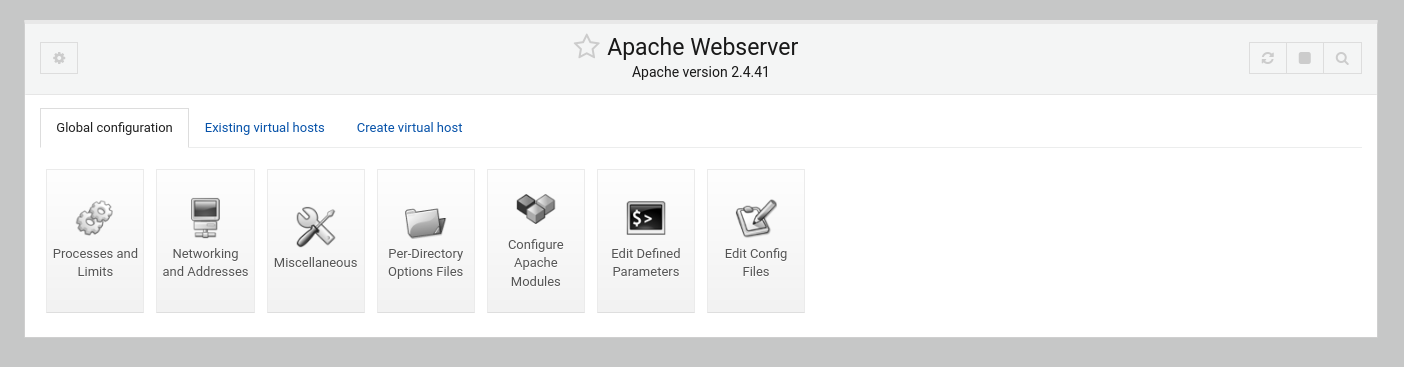
Los módulos se guardan en la configuración de Apache en dos directorios:

* ***/etc/apache2/mods-available/****:* Directorio que contiene los módulos disponibles en la instalación actual.
* ***/etc/apache2/mods-enabled/****:* Directorio que incluye mediante enlaces simbólicos al directorio anterior, los módulos que se van a cargar en memoria la próxima vez que se inicie Apache.

### ¿Cómo funciona y se configura el módulo UserDir?:

En sistemas con múltiples usuarios, cada usuario puede tener un sitio web en su directorio home usando el módulo UserDir.

Los visitantes de una URL “*http://<IP o dominio>/<nombre de usuario>/*” recibirán el contenido del directorio /home/<usuario>/public\_html del usuario.



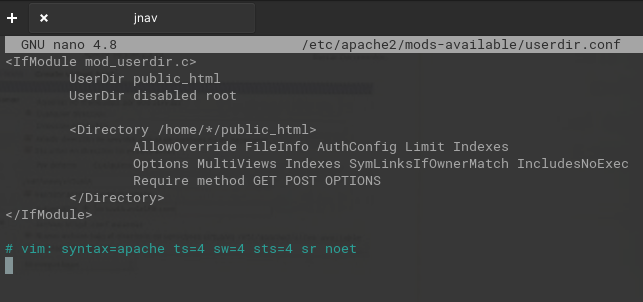
Para activar el módulo UserDir nos iremos al servidor web Apache en Webmin, nos iremos al apartado de configuración global y configuración de los módulos Apache:

Nos encontraremos con una gran lista de módulos, en específico buscaremos el módulo UserDir y clicaremos sobre su checkbox para activarlo:



Una vez activado, el módulo UserDir puede ser configurado desde la interfaz gráfica de Webmin o editando directamente su archivo de configuración.

Desde Webmin se haría directamente configurando el host virtual, pero desde el archivo de configuración, se haría editando el fichero */etc/apache2/mods-available/userdir.conf*.



En este fichero lo más importante, es que podemos editar donde queremos que se encuentre el directorio donde el usuario va a alojar su página web, por defecto es el directorio */home/usuario/public\_html*.

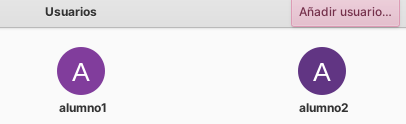
Si queremos cambiarlo, deberemos modificar la segunda y quinta línea indicando la nueva ruta.

Para poner en uso el módulo UserDir, primero debemos crear los usuarios en los que queremos albergar una página web si es que no los tenemos creados ya.

Entonces, para esta práctica, crearemos dos usuarios, alumno1 y alumno2. Los podemos crear rápidamente con el comando:

* **sudo adduser alumno1 && sudo adduser alumno2**

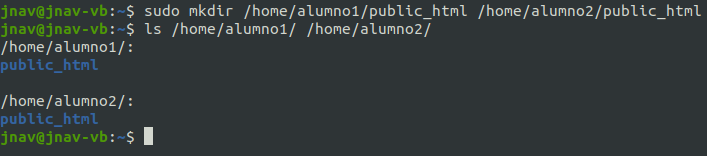




Cuando hayamos creado los dos usuarios de prueba, crearemos el directorio ***public\_html/*** en el home de cada usuario.

Para crearlos de manera rápida, podemos usar el siguiente comando:

* **sudo mkdir */home/alumno1/public\_html /home/alumno2/public\_html***

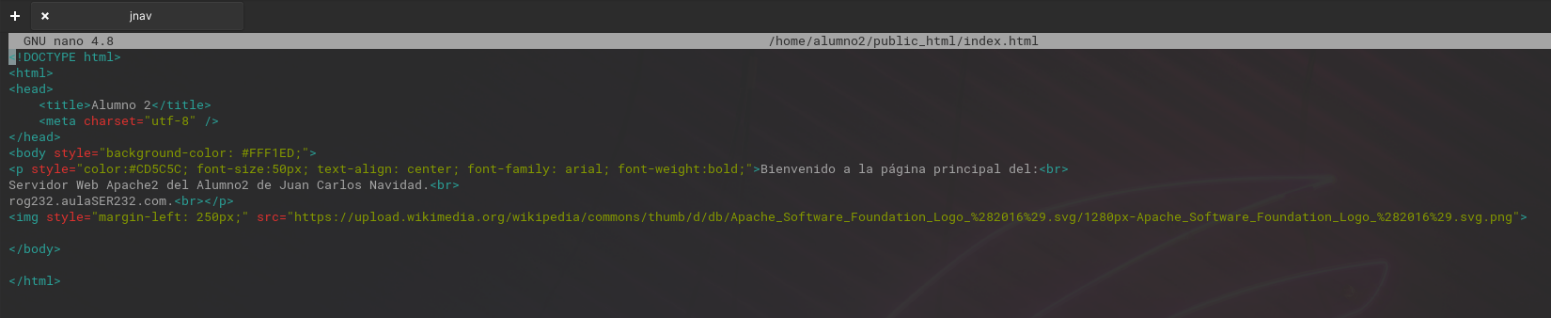


Una vez creados los directorios ***public\_html/***, vamos a añadirles una **página HTML** a cada uno llamada **index.html**. En mi caso he utilizado la plantilla utilizada para el sustituir la **página principal de Apache**, pero si queremos crearla de cero, se puede crear con el siguiente comando:

* **sudo nano */home/alumno1/public\_html/index.html***

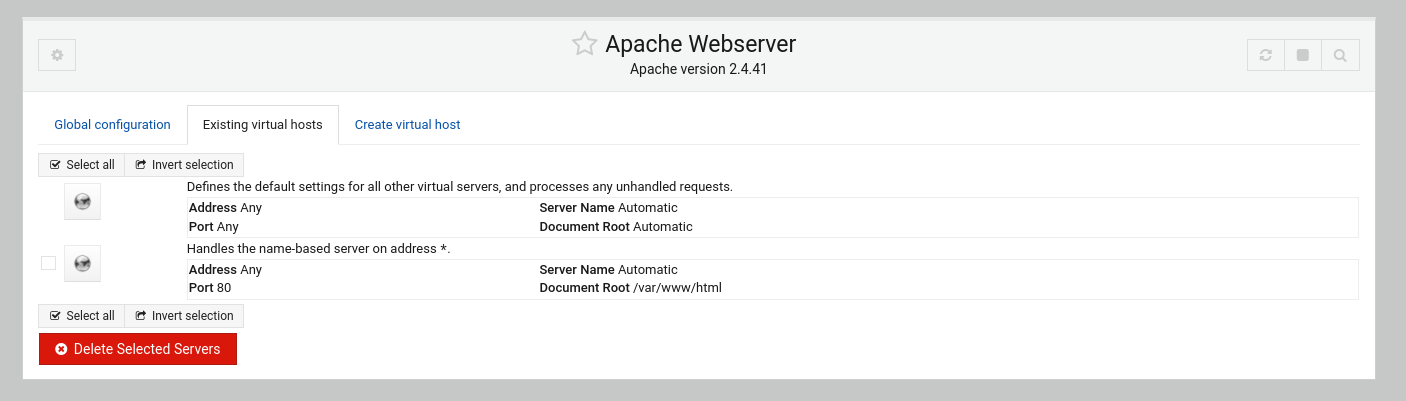
Se nos abrirá el editor de texto **nano** y ahí introduciremos el **código HTML**, mi código para cada uno de los usuarios es el siguiente:



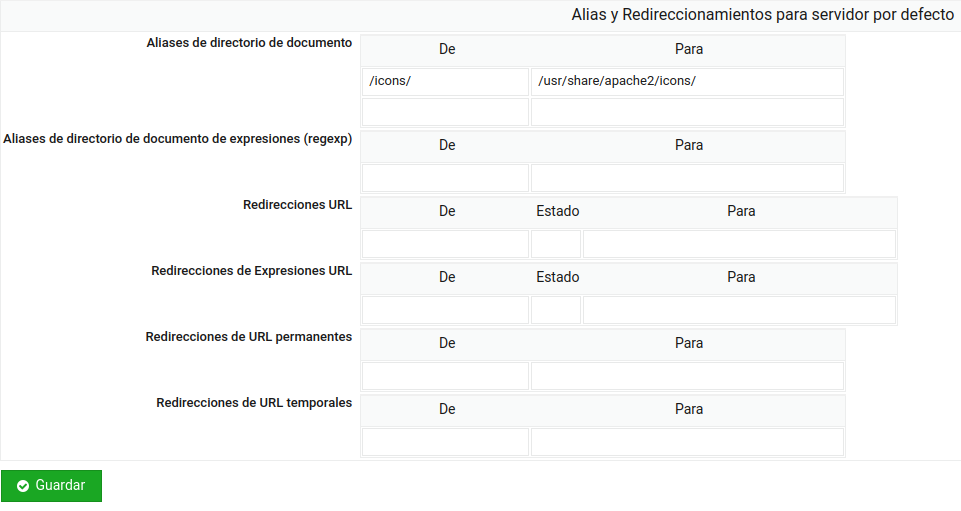


Después de tener creadas los **index.html** o, mejor dicho, las páginas que se van a mostrar en cada uno de los usuarios, debemos configurar el **host virtual** para que muestre nuestras páginas al introducir nuestro **servidor** o **IP** en el navegador.

Para eso, nos iremos a **Apache Webserver** dentro de **Webmin**, en el apartado **Existing virtual hosts**, clicaremos sobre el **primer host virtual** que aparece:

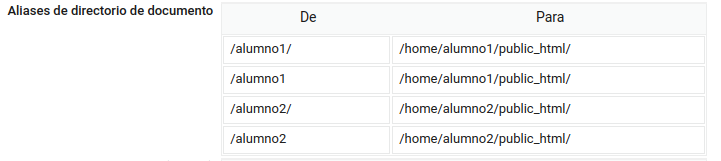


Nos aparecerá la siguiente pantalla:



De esta pantalla solamente utilizaremos el **primer apartado**, **Aliases de directorio de documento**. En el cuál borraremos la **primera entrada**: ***/icons/ /usr/share/apache2/icons/***.

Una vez eliminada, escribiremos lo siguiente:



Introduciremos los usuarios por duplicado, para que cuando introduzcamos la URL, acepte tanto **ip/alumnoX** como **ip/alumnoX/**, a continuación, se podrá ver en las capturas.

Una vez rellenados estos campos, le daremos a ***Guardar*** y a ***Aplicar cambios***.

### Comprobación de funcionamiento:

Después de haber realizado todos los pasos anteriores de manera correcta, debería de funcionarnos a la perfección, así que, para ponerlo a prueba, comprobaremos que funciona, introduciendo las **URLs** desde el navegador:





Incluso si tenemos activo el **servidor DNS** y nuestra **dirección IP** tiene asignada un **dominio**, podemos introducir el **dominio más /usuario**:



No se observa, pero si yo introduzco tanto ***192.168.103.232/alumno1*** como si introduzco ***192.168.103.232/alumno1/***, me carga la página igualmente.

### ¿Qué son los ficheros .htaccess?:

El archivo **.htaccess** (**acceso de hipertexto**) es un archivo oculto que se utiliza para configurar funciones adicionales para sitios web alojados en el **servidor web Apache**.

Permiten a los usuarios que no tienen permisos para modificar la configuración, de esa manera poder ejercer algún control sobre el comportamiento de su parte del servidor.

Entre lo que pueden hacer, puede **reescribir la URL**, **proteger directorios con contraseña**, **habilitar la protección de enlaces directos**, **no permitir el acceso a direcciones IP específicas**, **cambiar la zona horaria de tu sitio web** o **alterar la página de índice predeterminada**, y mucho más.

Se debe situar en el directorio al que tiene que afectar, junto a todos sus subdirectorios, y las modificaciones introducidas que no requieren reiniciar el servidor web.

### ¿Cómo crear ficheros .htaccess?:

Hay dos maneras para crearlos, podemos crearlos manualmente o mediante la interfaz gráfica de Webmin.

Para crearlo manualmente, abre tu editor de texto favorito y ahora sólo necesitas saber que código agregar, pondré algún que otro código que puede ser útil:

* **Bloquear IPs**

Order Deny, Allow

Deny from ip1

Deny from ip2

* **Bloquear acceso a archivos del servidor**

Order deny,allow

Deny from all

<Files ~ “.(xml|css|js|jpeg|png|gif|pdf|zip|rar)$”>

Allow from all

</Files>

* **Redirecciones**

RewriteEngine on

RewriteCond %{HTTP\_HOST} !^$

RewriteCond %{HTTP\_HOST} !^www\. [NC]

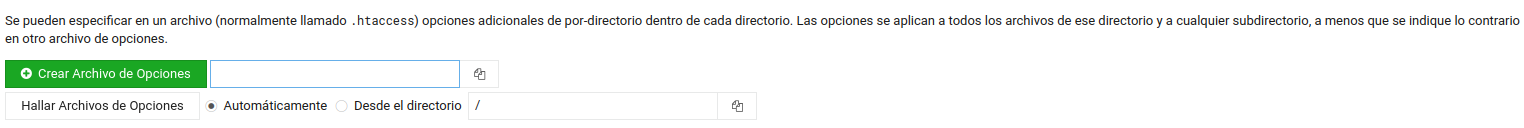
RewriteCond %{HTTPS}s ^on(s)|

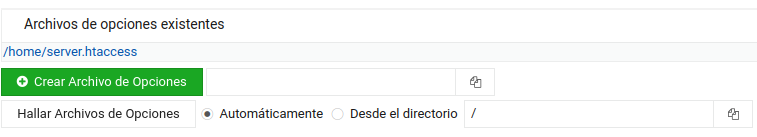
RewriteRule ^ http%1://www.%{HTTP\_HOST}%{REQUEST\_URI} [R=301,L]

Para crearlo desde la interfaz gráfica de Webmin debemos irnos dentro del Servidor Web Apache al apartado Configuración Global, y dentro de este apartado, clicar sobre *Archivos de Opciones de Por-Directorio*.

Dentro de este apartado, en la primera línea debemos indicar la ruta donde se va a hacer efectivo el archivo .htaccess.

La segunda línea nos da la opción de poder importar un archivo .htaccess creado manualmente, si le clicamos en el botón, los buscará automáticamente, también podemos introducir la ruta directamente.



Si por ejemplo creamos o importamos un archivo .htaccess, este a continuación nos aparecerá más arriba dentro del mismo apartado:

Si clicamos sobre el fichero .htaccess, podremos configurarlo con una gran variedad de opciones:



### Bibliografía:

1. [**¿Qué es HTTP y HTTPS?:**](#_¿Qué_es_HTTP)

<https://neo.lcc.uma.es/evirtual/cdd/tutorial/aplicacion/http.html>

<https://www.webempresa.com/hosting/que-es-servidor-web.html>

<https://developer.mozilla.org/es/docs/Web/HTTP/Overview>

1. [¿Qué es Apache?:](#_¿Qué_es_Apache?:)

<https://www.hostinger.es/tutoriales/que-es-apache/>

1. [¿Qué son los módulos Apache?:](#_¿Qué_son_los)

<http://luismido.wikidot.com/modulos-en-apache-2>

<https://plataforma.josedomingo.org/pledin/cursos/apache24/curso/u20/>

1. [¿Qué son los ficheros .htaccess?:](#_¿Qué_son_los_1)

Libro

<https://www.hostinger.es/tutoriales/que-es-el-archivo-htaccess>

1. [¿Cómo crear ficheros .htaccess?:](#_¿Cómo_crear_ficheros)

<https://www.hostinger.es/tutoriales/que-es-el-archivo-htaccess>