

PUESTA EN PRODUCCIÓN SEGURA

Práctica Apache

Parte 4 del Proyecto

(19-04-2024)

Rubén de la Viuda, David Sánchez y Hori Belchi

Ciberseguridad

Índice

1. Partiendo del archivo VagrantFile que hay en la carpeta Archivos de la unidad arranca la máquina virtual con apache ya instalado (crear un directorio, colocar el archivo anterior dentro y ejecutar vagrant up):.....	3
2. Nos conectamos a la máquina virtual anterior (en una terminal): vagrant ssh:..	3
3. Comprobamos que apache funciona:.....	4
4. Lista de comandos útiles:.....	5
5. Revisamos los archivos de configuración:.....	6
6. Servidor virtual por defecto:.....	6
7. Directivas:.....	8
8. Ficheros y directorios de prueba:.....	9
9. Opciones sobre directorios Directiva DirectoryIndex:.....	11
10. Opciones sobre directorios. Directivas Directory y Options Indexes:.....	12
11. Códigos del error. Directiva ErrorDocument:.....	13
12. Directorios Virtuales. Directiva Alias:.....	14
13. Redirecciones. Directiva Redirect:.....	15
14. Módulos:.....	15
15. Directivas Order, allow y deny. Directiva Require:.....	18
16. Autenticación y autorización Basic y Digest:.....	19
17. Fichero htaccess:.....	20
18. Módulos mod_status y mod_info:.....	20
19. Ficheros de registros (logs):.....	22
20. SSL en Apache:.....	23
21. Recomendaciones para securizar el servidor:.....	23

1. Partiendo del archivo VagrantFile que hay en la carpeta Archivos de la unidad arranca la máquina virtual con apache ya instalado (crear un directorio, colocar el archivo anterior dentro y ejecutar vagrant up):

(No hace falta capturas ni información para este apartado)

2. Nos conectamos a la máquina virtual anterior (en una terminal): vagrant ssh:

Muestra un pantallazo de la consola anterior:

```
ciber1@pcciber-06:~/PEPS/apache/practica$ vagrant ssh
Welcome to Ubuntu 14.04.6 LTS (GNU/Linux 3.13.0-170-generic x86_64)

* Documentation:  https://help.ubuntu.com/

System information as of Wed Feb 28 11:13:17 UTC 2024

System load:  0.07               Processes:            94
Usage of /:   3.7% of 39.34GB    Users logged in:     0
Memory usage: 6%               IP address for eth0: 10.0.2.15
Swap usage:   0%               IP address for eth1: 192.168.56.20

Graph this data and manage this system at:
https://landscape.canonical.com/

New release '16.04.7 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

-bash: warning: setlocale: LC_ALL: cannot change locale (es_ES.UTF-8)

WARNING! Your environment specifies an invalid locale.
This can affect your user experience significantly, including the
ability to manage packages. You may install the locales by running:

    sudo apt-get install language-pack-es
    or
    sudo locale-gen es_ES.UTF-8

To see all available language packs, run:
    apt-cache search "^language-pack-[a-z][a-z]$"
To disable this message for all users, run:
    sudo touch /var/lib/cloud/instance/locale-check.skip

-bash: warning: setlocale: LC_ALL: cannot change locale (es_ES.UTF-8)
-bash: warning: setlocale: LC_ALL: cannot change locale (es_ES.UTF-8)
vagrant@serverapachepractica:~$
```

Crea el usuario alumno, profesor, profesor1 y profesor2 . Indica la línea de comandos:

```
vagrant@serverapachepractica:~$ sudo useradd alumno
vagrant@serverapachepractica:~$ sudo useradd profesor1
vagrant@serverapachepractica:~$ sudo useradd profesor2
vagrant@serverapachepractica:~$ sudo mkdir /home/alumno
vagrant@serverapachepractica:~$ sudo mkdir /home/profesor1
vagrant@serverapachepractica:~$ sudo mkdir /home/profesor2
vagrant@serverapachepractica:~$ sudo chown alumno /home/alumno/
vagrant@serverapachepractica:~$ sudo chgrp alumno /home/alumno/
vagrant@serverapachepractica:~$ sudo chown profesor1 /home/profesor1/
vagrant@serverapachepractica:~$ sudo chgrp profesor1 /home/profesor1/
vagrant@serverapachepractica:~$ sudo chown profesor2 /home/profesor2/
vagrant@serverapachepractica:~$ sudo chgrp profesor2 /home/profesor2/
vagrant@serverapachepractica:~$ █
```

3. Comprobamos que apache funciona:

Ejecutar el siguiente comando para ver el proceso principal de apache: ps -ef | grep apache2:

```
vagrant@serverapachepractica:~$ ps -ef | grep apache2
root      1230      1  0 11:13 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1233    1230  0 11:13 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1234    1230  0 11:13 ?        00:00:00 /usr/sbin/apache2 -k start
vagrant   2144    2052  0 11:17 pts/0    00:00:00 grep --color=auto apache2
vagrant@serverapachepractica:~$ █
```

Ejecutar el siguiente comando para ver los puertos en escucha: netstat -ltn:

```
vagrant@serverapachepractica:~$ netstat -ltn
Active Internet connections (only servers)

```

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	0.0.0.0:111	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:22	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:41079	0.0.0.0:*	LISTEN
tcp6	0	0	:::111	:::*	LISTEN
tcp6	0	0	:::80	:::*	LISTEN
tcp6	0	0	:::22	:::*	LISTEN
tcp6	0	0	:::37514	:::*	LISTEN

```
vagrant@serverapachepractica:~$ █
```

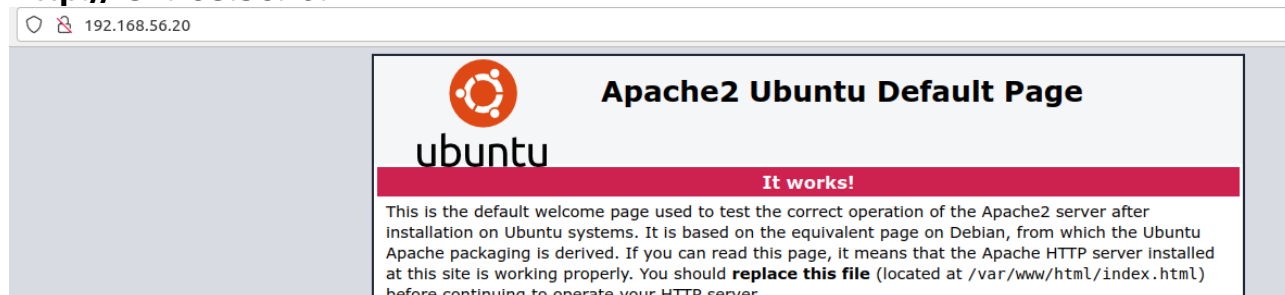
Ejecutar el siguiente comando para ver el contenido del directorio donde se colocarán los archivos que se van a publicar: ls -l /var/www/html:

```
vagrant@serverapachepractica:~$ ls -l /var/www/html
total 12
-rw-r--r-- 1 root root 11510 Feb 28 11:08 index.html
vagrant@serverapachepractica:~$ █
```

Ejecutar el siguiente comando para ver con que versión está funcionando apache: `apache2 -v`:

```
vagrant@serverapachepractica:~$ apache2 -v
Server version: Apache/2.4.7 (Ubuntu)
Server built:   Apr  3 2019 18:04:25
vagrant@serverapachepractica:~$
```

Abrir un navegador y accedemos al servidor web anterior con la url: `http://192.168.56.10`:



Modifica el archivo `/var/www/html/index.html` y comprueba en el navegador que se ven los cambios:



4. Lista de comandos útiles:

(No hace falta capturas ni información para este apartado)

5. Revisamos los archivos de configuración:

Acceder al directorio /etc/apache2. Hacer un listado del directorio e indica los ficheros de configuración:

```
vagrant@serverapachepractica:~$ cd /etc/apache2
vagrant@serverapachepractica:/etc/apache2$ ls -l
total 80
-rw-r--r-- 1 root root 7115 Nov 26 2018 apache2.conf
drwxr-xr-x 2 root root 4096 Feb 28 11:08 conf-available
drwxr-xr-x 2 root root 4096 Feb 28 11:08 conf-enabled
-rw-r--r-- 1 root root 1782 Nov 26 2018 envvars
-rw-r--r-- 1 root root 31063 Nov 26 2018 magic
drwxr-xr-x 2 root root 12288 Feb 28 11:08 mods-available
drwxr-xr-x 2 root root 4096 Feb 28 11:08 mods-enabled
-rw-r--r-- 1 root root 320 Nov 26 2018 ports.conf
drwxr-xr-x 2 root root 4096 Feb 28 11:08 sites-available
drwxr-xr-x 2 root root 4096 Feb 28 11:08 sites-enabled
vagrant@serverapachepractica:/etc/apache2$ █
```

Los archivos de configuración son los que incluyen 'conf' en su nombre.

6. Servidor virtual por defecto:

Accede al directorio /etc/apache2/sites-available y comprueba que está creado el archivo 000-default. Este fichero contiene la configuración del servidor virtual por defecto. Lee brevemente su contenido. Explica que crees que se está estableciendo:

Comprobamos que está creado:

```
vagrant@serverapachepractica:~$ cd /etc/apache2/sites-available
vagrant@serverapachepractica:/etc/apache2/sites-available$ ls -l 000-default.conf
-rw-r--r-- 1 root root 1332 Nov 26 2018 000-default.conf
vagrant@serverapachepractica:/etc/apache2/sites-available$ █
```

Su contenido es el siguiente:

```
vagrant@serverapachepractica:/etc/apache2/sites-available$ cat 000-default.conf

<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
vagrant@serverapachepractica:/etc/apache2/sites-available$ █
```

Está estableciendo un servidor virtual que escuche en el puerto 80 de todas sus direcciones IP. Establece el administrador del servidor virtual (webmaster de la propia máquina), cuál es la raíz de la página y cuáles son los directorios para almacenar los logs.

Accede a `/etc/apache2/sites-enabled` y comprueba si existe dicho fichero `000-default.conf`. ¿Existe? ¿Que aparece?

El directorio `/etc/apache2` es el directorio raíz de configuración de Apache. Almacena toda la configuración relacionada con la aplicación.

Explica que relación existe entre los ficheros y directorios mencionados (/etc/apache2, /etc/apache2/apache2.conf, /etc/apache2/sites-available y /etc/apache2/sites-enabled)

El archivo /etc/apache2/apache2.conf es el archivo de configuración general de Apache. Cualquier configuración que se especifique en este archivo se aplicará a todo Apache.

El directorio /etc/apache2/sites-available es el directorio donde se almacenan todas las configuraciones individuales para cada uno de los sitios web que tengamos en el servidor.

El directorio /etc/apache2/sites-enabled es el directorio donde se especifica qué sitios web están activos y deben ser tenidos en cuenta. Sus archivos son simples enlaces a los archivos de configuración en /etc/apache2/sites-available de los sitios web activos.

7. Directivas:

Consulta la documentación de Apache y responde a las siguientes preguntas:

¿Se permiten conexiones persistentes (que todas las conexiones de un usuario se atienden en la misma conexión TCP)? ¿Qué directiva define este comportamiento?

Sí, es la directiva: KeepAlive On

¿Cuál es el fichero de errores? ¿Qué directiva lo define?

Error.log, lo define ErrorLog

Consulta el fichero /etc/apache2/ports.conf y comprueba cual es el puerto en el que escucha peticiones el servidor Apache. ¿En qué puerto escuchará también si se habilita el módulo ssl?

Ahora mismo escucha el puerto 80, si activas el módulo ssl también escucha en el 443

Consulta el fichero /etc/apache2/sites-available/000-default y observa las líneas contenidas entre las directivas <VirtualHost> y </VirtualHost> El valor de la directiva DocumentRoot es /var/www/html. ¿Para qué crees que sirve esta línea?

Es la directiva que indica en que carpeta están los ficheros que mostrará apache.
Observa la directiva Errorlog \$APACHE_LOG..... ¿Para qué crees que sirve esta línea?

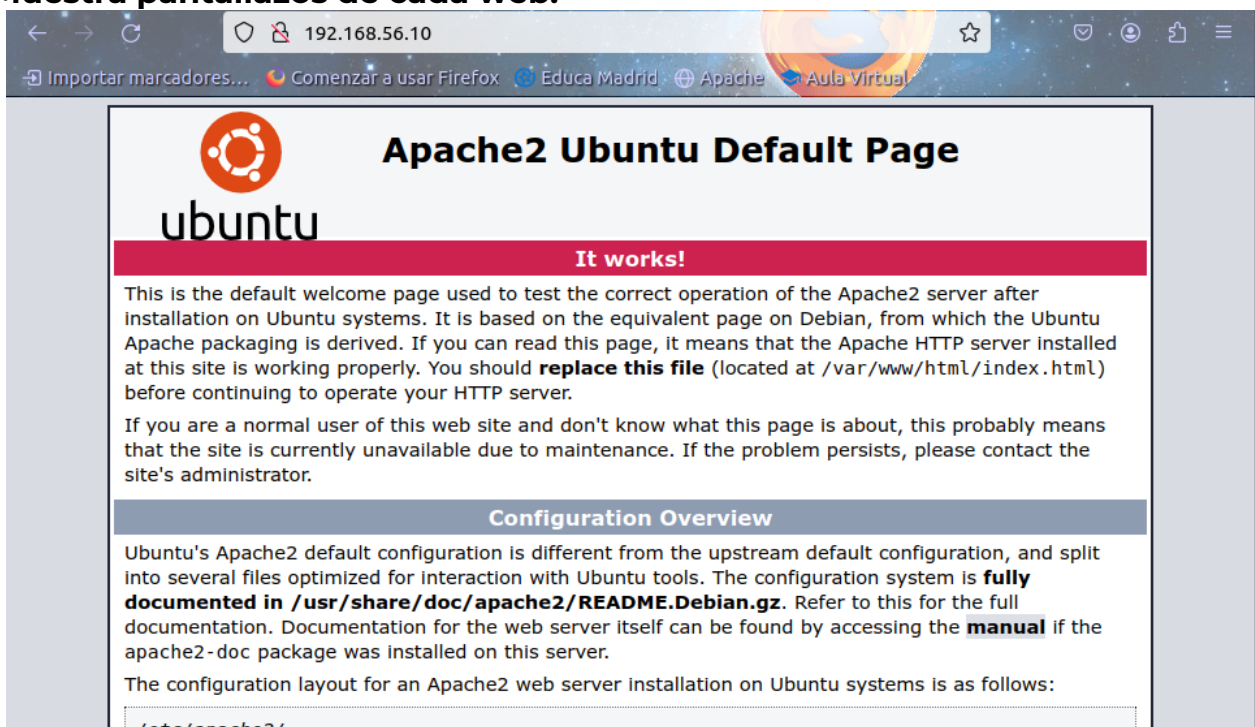
Sirve para indicar en qué log se guardan los errores producidos en apache (para los ofrecidos por 000-default)

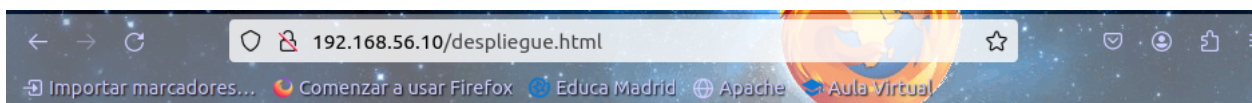
Consulta el fichero /etc/apache2/apache2.conf. Aparecen varias secciones enmarcadas por las directivas <Directory> ...</Directory>. ¿Para qué crees que sirven estas líneas?

Son las directivas que marcan el funcionamiento y los permisos de los ficheros y directorios ofrecidos por 000-default

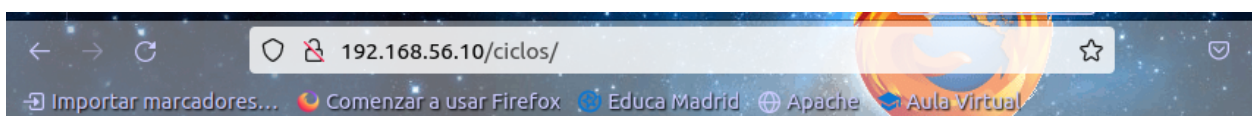
8. Ficheros y directorios de prueba:

Muestra pantallazos de cada web:










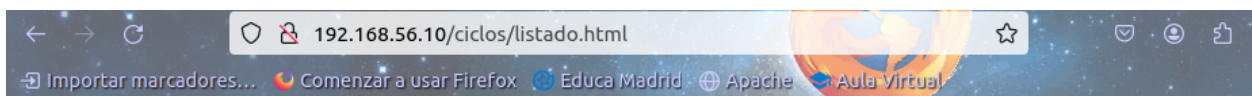
Despliegue Grupo 3 (Hori,RubÃ©n de la Viuda y David SÃ¡nchez)



Index of /ciclos

Name	Last modified	Size	Description
<hr/>			
 Parent Directory		-	
 asir.html	2024-02-28 11:22	72	
 dam.html	2024-02-28 11:23	72	
 daw.html	2024-02-28 11:23	72	
 listado.html	2024-02-28 11:23	69	

Apache/2.4.7 (Ubuntu) Server at 192.168.56.10 Port 80



Listado Grupo 3 (Hori,RubÃ©n de la Viuda y David SÃ¡nchez)

9. Opciones sobre directorios Directiva DirectoryIndex:

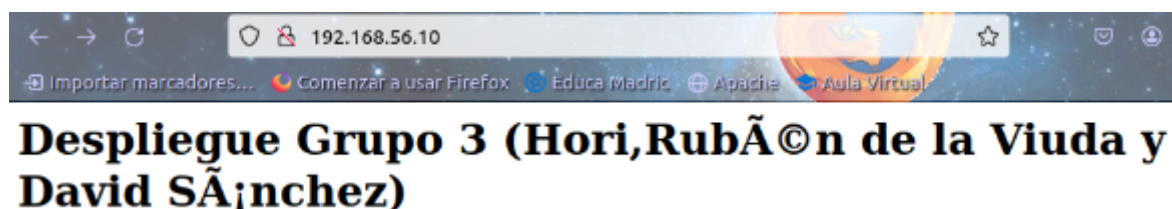
Desde Cliente abre un navegador y establece una conexión a <http://192.168.56.10>

Renombra el fichero `/var/www/html/index.html` a `/var/www/html/indice.html` y vuelve a establecer una conexión a <http://192.168.56.10> ¿Cuál es el resultado? ¿Qué se muestra en pantalla? ¿Por qué?



Muestra el directorio ya que apache está configurado para mostrar el contenido de "index.html" por defecto al conectarte a la ip, pero al no existir, te muestra todos los archivos.

Edita el archivo `/etc/apache2/sites-available/000-default.conf` y crea la sección `<Directory /var/www/html> ... </Directory>`. Añade la directiva `DirectoryIndex` con el valor `despliegue.html`. Reinicia el servidor para que los cambios tengan efecto. Vuelve a establecer una conexión con <http://192.168.56.10> ¿Que aparece en pantalla? ¿Por qué?








Aparece el contenido de despliegue, ya que la directiva usada indica que fichero usar como el mostrado por defecto al intentar ver el contenido de `/var/www/html` (La carpeta por defecto de apache2)

10. Opciones sobre directorios. Directivas Directory y Options Indexes:

Desde Cliente abre un navegador y establece una conexión a <http://192.168.56.10/ciclos> . ¿Que aparece en nuestro navegador? ¿Por qué?



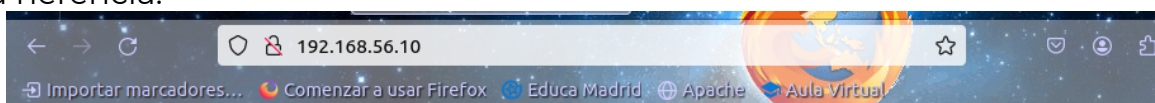
Index of /ciclos

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 asir.html	2024-02-28 11:22	72	
 dam.html	2024-02-28 11:23	72	
 daw.html	2024-02-28 11:23	72	
 listado.html	2024-02-28 11:23	69	

Apache/2.4.7 (Ubuntu) Server at 192.168.56.10 Port 80

Aparece el sistema de archivos del directorio, ya que no hay configuraciones específicas para el directorio ciclos.

Observa que para `/var/www/html/ciclos` no se ha definido la opción Indexes . Reinicia el servidor y desde Cliente abre un navegador y establece una conexión a <http://192.168.56.10/> . ¿Que aparece en pantalla? ¿Por qué? Explica cómo funciona esta herencia:



Despliegue Grupo 3 (Hori,Rubén de la Viuda y David SÁnchez)

Esto muestra despliegue.html, ya que coge la herencia de `/var/www/html` y tiene como default a `despliegur.html`

Desde Cliente abre un navegador y establece una conexión a <http://192.168.56.10/ciclos> . ¿Que aparece en pantalla? ¿Por qué? Explica cómo funciona esta herencia:



Forbidden

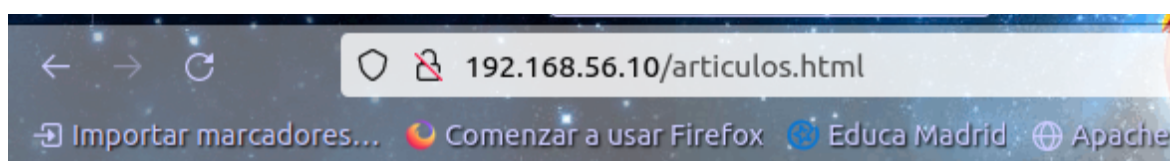
You don't have permission to access /ciclos/ on this server.

Apache/2.4.7 (Ubuntu) Server at 192.168.56.10 Port 80

En este caso está forbidden, ya que coge los parámetros de /var/www/html/ciclos y estos prohíben el acceso a un directorio para ver los archivos de dentro, aunque si permite ver los propios archivos si los pones en la url.

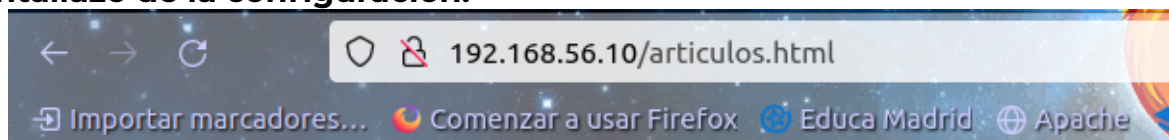
11. Códigos del error. Directiva ErrorDocument:

Configura el servidor virtual por defecto para que cuando retorne el código de error 404 (pagina no encontrada) envíe el texto "Página no encontrada en el servidor de la red NombreDeTuServidor ". Reinicia el servidor para que los cambios tengan efecto. Desde Cliente abre un navegador y establece una conexión a <http://192.168.56.10/articulos.html>. ¿Qué aparece en pantalla?:



Página no encontrada en el servidor de la red Grupo3 CiberA

Configura el servidor virtual por defecto para que cuando retorne el código de error 404 (página no encontrada) envíe la página 404.html almacenada en el directorio raíz del servidor (Crea el fichero /var/www/html/404.html con el contenido que quieras). Comprueba que funciona correctamente. Muestra pantallazo de la configuración:

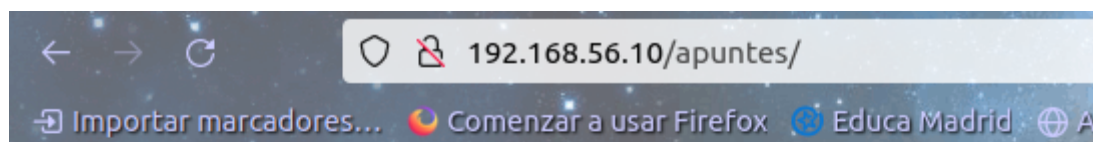


Página de error 404 Grupo3

```
</Directory>
ErrorDocument 404 /404.html
# Available loglevels: trace8, ..., t
```

12. Directorios Virtuales. Directiva Alias:

Desde Cliente abre un navegador y establece una conexión a <http://192.168.56.10/apuntes>
Muestra un pantallazo:



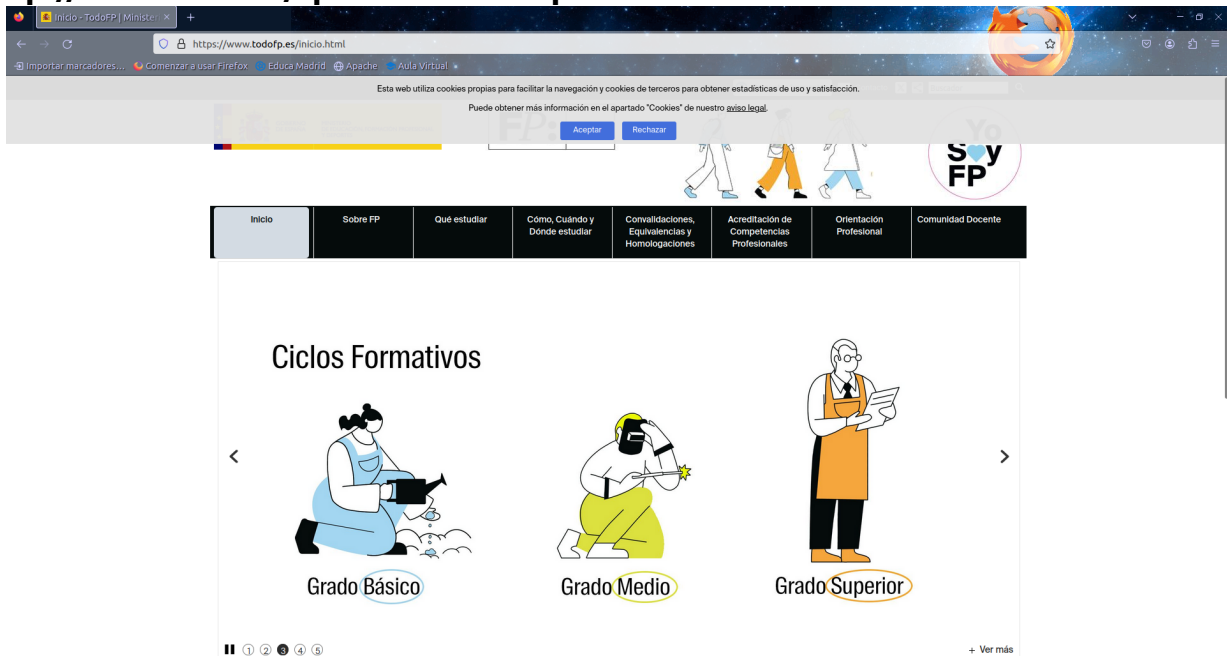
Index of /apuntes

Name	Last modified	Size	Description
Parent Directory		-	

Apache/2.4.7 (Ubuntu) Server at 192.168.56.10 Port 80

13. Redirecciones. Directiva Redirect:

Desde Cliente abre un navegador y establece una conexión a <http://192.168.56.10/fp>. Muestra un pantallazo:



14. Módulos:

Consulta de Módulos cargados

Comprueba los módulos estáticos que se han cargado al compilar el servidor.
¿Qué comando utilizamos? ¿Cuáles son los módulos?

apache2 -M

```
vagrant@serverapache:/$ apache2 -M
[Thu Apr 04 07:13:23.722599 2024] [core:warn] [pid 3976] AH00111: Config variable
e ${APACHE_LOCK_DIR} is not defined
[Thu Apr 04 07:13:23.722645 2024] [core:warn] [pid 3976] AH00111: Config variabl
e ${APACHE_PID_FILE} is not defined
[Thu Apr 04 07:13:23.722658 2024] [core:warn] [pid 3976] AH00111: Config variabl
e ${APACHE_RUN_USER} is not defined
[Thu Apr 04 07:13:23.722665 2024] [core:warn] [pid 3976] AH00111: Config variabl
e ${APACHE_RUN_GROUP} is not defined
[Thu Apr 04 07:13:23.722679 2024] [core:warn] [pid 3976] AH00111: Config variabl
e ${APACHE_LOG_DIR} is not defined
[Thu Apr 04 07:13:23.723674 2024] [core:warn] [pid 3976:tid 140220407609216] AH0
0111: Config variable ${APACHE_LOG_DIR} is not defined
[Thu Apr 04 07:13:23.723747 2024] [core:warn] [pid 3976:tid 140220407609216] AH0
0111: Config variable ${APACHE_LOG_DIR} is not defined
[Thu Apr 04 07:13:23.723757 2024] [core:warn] [pid 3976:tid 140220407609216] AH0
0111: Config variable ${APACHE_LOG_DIR} is not defined
AH00526: Syntax error on line 74 of /etc/apache2/apache2.conf:
Invalid Mutex directory in argument file:${APACHE_LOCK_DIR}
```

Comprueba los módulos dinámicos que se han cargado al arrancar el servidor. ¿Qué comando utilizamos? ¿Que obtenemos? ¿Cuáles son los módulos?:

apache2ctl -M | Obtenemos una lista de los módulos dinámicos:

```
vagrant@serverapache:/$ apache2ctl -M
Loaded Modules:
core_module (static)
so_module (static)
watchdog_module (static)
http_module (static)
log_config_module (static)
logio_module (static)
version_module (static)
unixd_module (static)
access_compat_module (shared)
alias_module (shared)
auth_basic_module (shared)
authn_core_module (shared)
authn_file_module (shared)
authz_core_module (shared)
authz_host_module (shared)
authz_user_module (shared)
autoindex_module (shared)
deflate_module (shared)
dir_module (shared)
env_module (shared)
filter_module (shared)
```

Edita uno de los ficheros .load (por ejemplo dir.load) y observa cómo se utiliza la directiva LoadModule para cargar el módulo. Comprueba cual es la ruta donde está el código del módulo (archivo .so):

/usr/lib/apache2/modules/

Consulta el directorio /usr/lib/apache2/modules/ y observa los módulos disponibles para cargar. Elige un par de módulos y busca en <http://httpd.apache.org/docs/2.4/es/mod/> su funcionalidad:

El módulo mod_unique_id.so sirve para generar un identificador único para cada solicitud que recibe el servidor.

El módulo mod_session.so proporciona funcionalidades para la gestión de sesiones de usuario

Ejecuta el comando “sudo apt-cache search libapache2-mod” para mostrar los paquetes disponibles en los repositorios de Ubuntu que permiten instalar módulos adicionales en Apache. Elige un paquete y busca documentación acerca de dicho paquete:

El paquete libapache2-mod-webauth es un módulo de apache que proporciona integración con el sistema de autenticación WebAuth

Módulo userdir

Habilita el módulo. (Indica el comando necesario):

`sudo a2enmod userdir`

Verifica que dentro del directorio `/etc/apache2/mods-enabled` se han creado enlaces simbólicos del módulo `userdir` hacia `/etc/apache2/mod-availables`. ¿Como se llaman los ficheros `.conf` y `.load` creados? Reinicia el servidor para que los cambios surtan efecto:

```
lrwxrwxrwx 1 root root 30 Apr  4 07:23 userdir.conf -> ../mods-available/userdir.conf
lrwxrwxrwx 1 root root 30 Apr  4 07:23 userdir.load -> ../mods-available/userdir.load
vagrant@serverapache:/etc/apache2/mods-enabled$
```

Consulta el fichero `/etc/apache2/mod enabled/userdir.conf` . ¿Qué significado tiene el contenido de dicho fichero?

El contenido del fichero permite el manejo de acceso a los directorios personales de los usuarios a través de `userdir`.

Indica los pasos a seguir para que aparezca en pantalla el contenido de un fichero denominado `personal.html` (con el contenido que quieras) cuando desde Cliente se accede a `http://192.168.56.10/~alumno`. Asegurate que el usuario `alumno` esta creado en la máquina.

Una vez está creado el usuario, creamos un directorio de inicio del usuario como por ejemplo `public_html` donde meteremos la página `personal.html` y le daremos los permisos correspondientes para que no de problemas, comprobamos que `userdir` está habilitado y si está todo bien con la URL `http://192.168.56.10/~alumno/personal.html` podremos visualizar la página

Deshabilita el módulo. Indica el comando a utilizar:

`sudo a2dismod userdir`

15. Directivas Order, allow y deny. Directiva Require:

En la máquina crea el directorio `/var/www/html/profesor` y crea en este directorio un fichero html con el contenido que quieras. Configura Apache para que solo se pueda acceder desde el equipo local y desde la máquina Cliente (utilizando su dirección IP). Muestra pantallazo:

```
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

<Directory /var/www/html/profesor>
Options Indexes FollowSymLinks
AllowOverride None
Require ip 192.168.56.10 127.0.0.1
</Directory>

# Available loglevels: trace8, ..., trace1, debug
```

Indica y prueba dos soluciones distintas, utilizando las directivas Order, Allow y Deny. Comprueba que se puede acceder a `http://192.168.56.10/profesor/` desde Cliente pero no desde otras máquinas de compañeros. Muestra pantallazo:

```
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

<Directory /var/www/html/profesor>
Options Indexes FollowSymLinks
AllowOverride None
Order deny,allow
Deny from all
Allow from 192.168.56.100
</Directory>

Get Help      ^O WriteOut    ^R Read File
Exit          ^J Justify     ^W Where Is
```

Vuelve a realizar los dos puntos anteriores pero esta vez utilizando la directiva `Require` en lugar de `Order`, `Allow` y `Deny`. Muestra pantallazo:

```
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

<Directory /var/www/html/profesor>
Options Indexes FollowSymLinks
AllowOverride None
Require local
Require ip 192.168.56.100
</Directory>

Help      ^O WriteOut      ^R Read File
t         ^L Justify       ^W Where Is
```

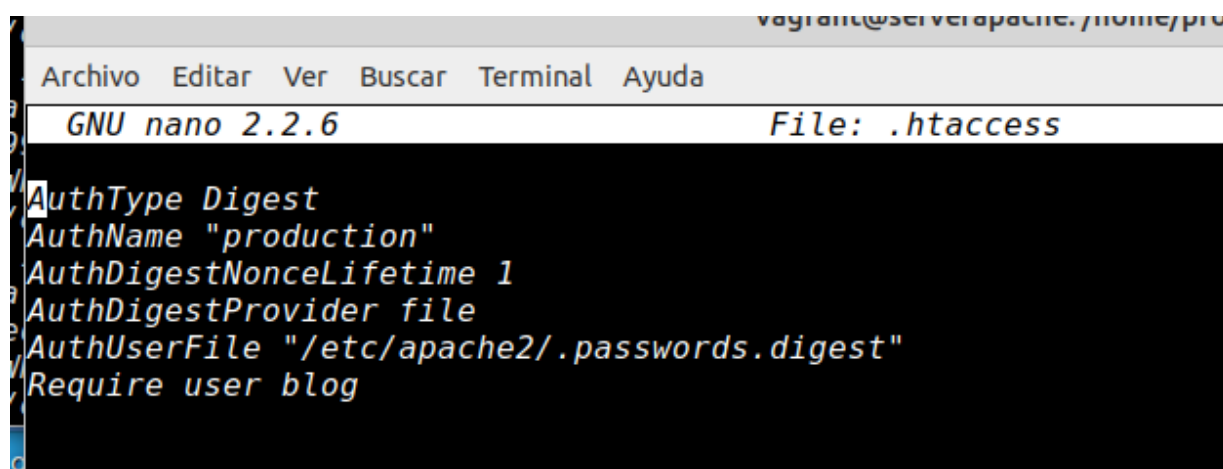
16. Autenticación y autorización Basic y Digest:

(No hace falta capturas ni información para este apartado)

17. Fichero htaccess:

Muestra pantallazo:

```
</Directory>
<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>
```



The screenshot shows a terminal window with the nano editor open. The title bar indicates the user is 'vagrant@serverapache' and the current directory is '/home/pro'. The menu bar includes 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The status bar shows 'GNU nano 2.2.6' and 'File: .htaccess'. The main content area displays the following configuration:

```
AuthType Digest
AuthName "production"
AuthDigestNonceLifetime 1
AuthDigestProvider file
AuthUserFile "/etc/apache2/.passwords.digest"
Require user blog
```

18. Módulos mod_status y mod_info:

Edita el fichero de configuración del módulo `/etc/apache2/mods-enabled/status.conf` y habilita el acceso a `/server-status` desde Cliente. Muestra pantallazo:

```
# Allow server status reports generated by mod_status
# with the URL of http://servername/server-status
# Uncomment and change the "192.0.2.0/24" to allow access from
# the desired network.
<Location /server-status>
    SetHandler server-status
    Require ip 192.168.56.100 127.0.0.1
</Location>

# Keep track of extended status information for
ExtendedStatus On

# Determine if mod_status should log the first 60
```

Desde Cliente conéctate a `http://192.168.56.10/server-status`. Muestra pantallazo:



Nota: No se porque no me deja acceder si esta bien configurado

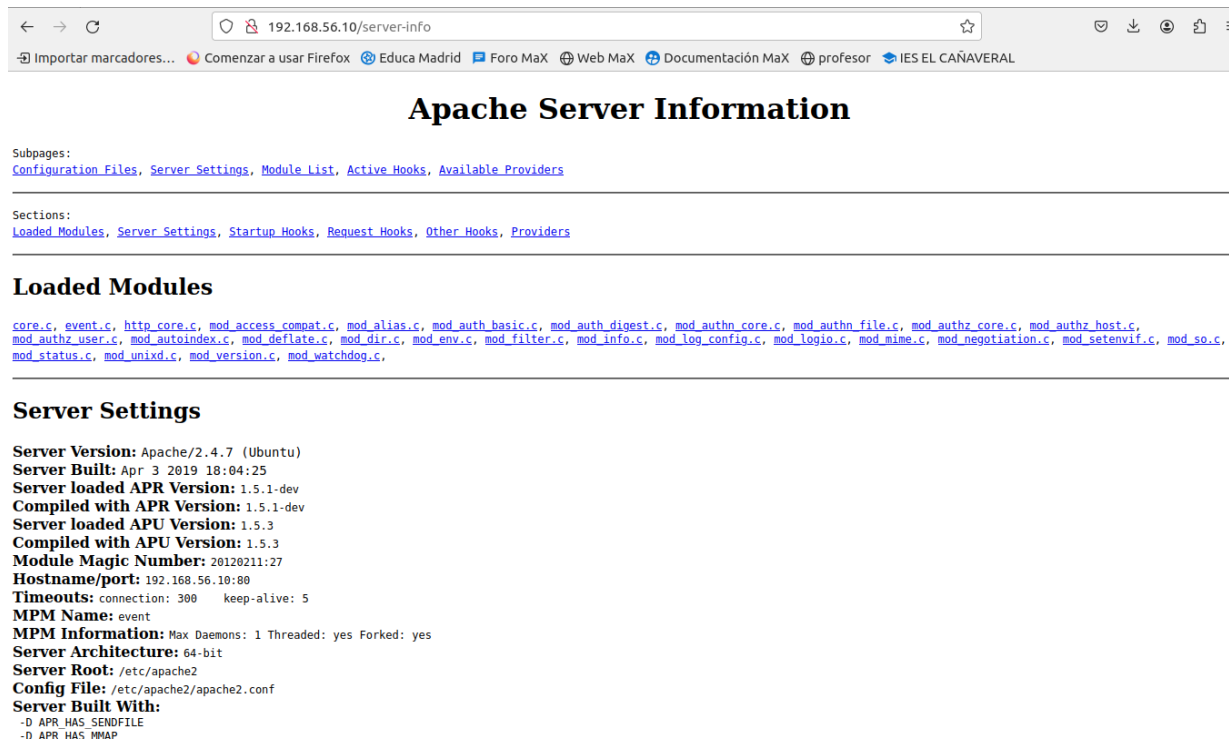
Edita el fichero de configuración del módulo `/etc/apache2/mods-enabled/info.conf` y habilita el acceso a `/server-info` desde Cliente. Muestra pantallazo:

```
# http://servername/server-info (requires
# Uncomment and change the "192.0.2.0/24"
#
<Location /server-info>
    SetHandler server-info
    Require ip 192.168.56.10/24
</Location>

odule>

syntax=apache ts=4 sw=4 sts=4 sr noet
```

Desde Cliente conéctate a <http://192.168.56.10/server-info>. Muestra pantallazo:



19. Ficheros de registros (logs):

¿Cuál es el fichero de logs de errores (directiva ErrorLog) y cuál es su nivel de prioridad (directiva LogLevel)?:

```
#LogLevel info ssl:warn  
ErrorLog ${APACHE_LOG_DIR}/error.log
```

Observa cual es el fichero de logs de accesos (directiva CustomLog) y cuál es su formato (como no especifica ningún formato con LogFormat se usa el predefinido para el servidor principal en el fichero /etc/apache2/apache2.conf):

```
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

20. SSL en Apache:

(No hace falta capturas ni información para este apartado)

21. Recomendaciones para securizar el servidor:

Permitir sólo determinados métodos de solicitud HTTP (LimitExcept en cada archivo site)

Por defecto, Apache acepta multitud de métodos de solicitud dentro del protocolo HTTP. Por ejemplo, suele aceptar protocolos como OPTIONS, GET, HEAD, POST, PUT, DELETE, TRACE, CONNECT.

Sin embargo, en la gran mayoría de las veces solo nos interesan los métodos POST y GET; mientras que el resto no los vamos a utilizar y mantenerlos activos puede romper la seguridad de nuestra aplicación web.

Para permitir únicamente los métodos POST y GET modificamos con nano el archivo `/etc/apache2/apache.conf` y nos movemos hasta el directorio que queramos modificar:

```
<Directory /home/alumno/apuntes/>  
    Options Indexes FollowSymLinks  
    AllowOverride None  
    Require all granted  
</Directory>
```

Una vez aquí, añadimos lo siguiente a su interior:

```
<Directory /home/alumno/apuntes/>  
    Options Indexes FollowSymLinks  
    AllowOverride None  
    Require all granted  
    Order allow,deny  
    Allow from all  
    <LimitExcept POST GET>  
        Deny from all  
    </LimitExcept>  
</Directory>
```

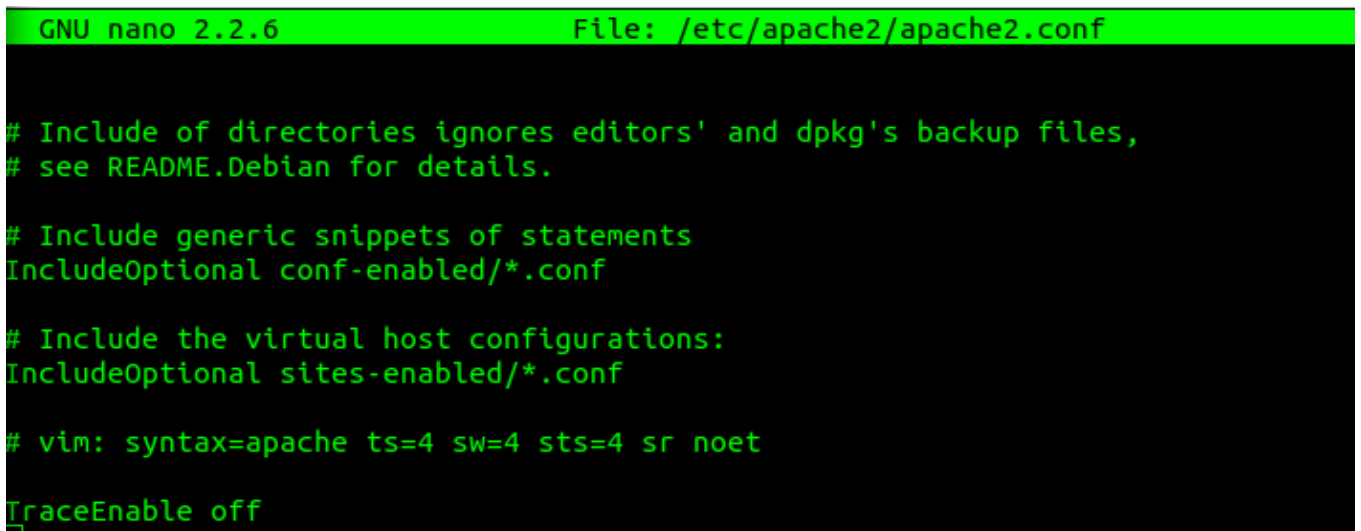
Guardamos y reiniciamos Apache:

```
vagrant@serverapache:~$ sudo service apache2 restart
* Restarting web server apache2
Apache needs to decrypt your SSL Keys for ciberIES:443 (RSA)
Please enter passphrase: [ OK ]
vagrant@serverapache:~$
```

Ya tenemos configurados los métodos de HTTP permitidos para dicho directorio. Ahora tendremos que hacer lo mismo para cada uno de los directorios que queramos proteger de esta forma.

Deshabilitar la solicitud HTTP de seguimiento (TraceEnable en apache2.conf)

La solicitud TraceEnable permite preguntar a apache2 sobre las acciones y el camino que realiza para responder la petición, su uso debería ser solo para debug y nunca en producción, ya que permite descubrir vulnerabilidades a los atacantes, para deshabilitarlo:



```
GNU nano 2.2.6 File: /etc/apache2/apache2.conf

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

TraceEnable off
```

Una vez escrito “TraceEnable off” en el archivo de configuración global de apache, reiniciamos el servicio, y ya tenemos TraceEnable desactivado.