

## Despliegue de Aplicaciones Web

### UD 2. Servidor HTTP

#### Práctica 2.2: Administración de Apache II Módulos



**DANIEL SEGURA VELASCO**  
**2º DAW**



- Crea un fichero que se llame Practica2.2\_Apellido1Apellido2\_Nombre.pdf .  
Inserta todas las capturas de pantallas por orden explicando cada una de ellas.
- Una vez terminada la práctica, sube el archivo.

## Contenido

### A) Módulos en Linux

#### A.1) Módulos

#### A.2) Módulo userdir

#### A.3) Módulo userdir en el servidor de clase

### B) Control de acceso por IP y nombre de dominio

### C) Autenticación y autorización Basic y Digest

#### C.1) Autenticación Basic

#### C.2) Autenticación Digest

### D) Ficheros .htaccess (si no sale poner pantallazo de haberlo intentado)

### E) Ficheros de registros (logs)

### F) Módulos status e info

### G) Webalizer

### H) GitHub

## A) Módulos en Linux

El servidor HTTP Apache es **MODULAR**, lo cual quiere decir que se pueden añadir módulos para darle otras funcionalidades al servidor HTTP. En este apartado vamos a ver como se cargan nuevos módulos y como se descargan dichos módulos en Linux y le daremos uso.

Existen módulos estáticos, que se cargan al compilar el servidor y se pueden ver mediante el comando:

```
sudo apache2ctl -l
```

También existen módulos dinámicos, los cuales pueden cargarse y descargarse de manera dinámica. En Linux, los módulos disponibles se encuentran en el directorio

```
/etc/apache2/mods-available/
```

Los archivos **.load** sirven para cargar el módulo y los **.conf** para configurarlo.

Mientras que los módulos que están cargados se encuentran en el directorio

```
/etc/apache2/mods-enabled/
```

Para habilitar y deshabilitar módulos se usan los comandos:

```
a2enmod nombre_del_modulo  
a2dismod nombre_del_modulo
```

Cada vez que se carga/descarga un módulo, tendrás que reiniciar el servidor Apache.

Los módulos existentes se pueden consultar en: <http://httpd.apache.org/docs/2.2/mod/>

## A.1) Módulos

**PASO 1)** Comprueba los módulos estáticos que se han cargado al compilar el servidor ejecutando el comando correspondiente.

**PASO 2)** Comprueba los módulos que se han cargado dinámicamente al arrancar el servidor.

**PASO 3)** Edita uno de los archivos `.load` y observa cómo se usa la directiva `LoadModule`. ¿Qué extensión tienen los archivos donde está el código del módulo?

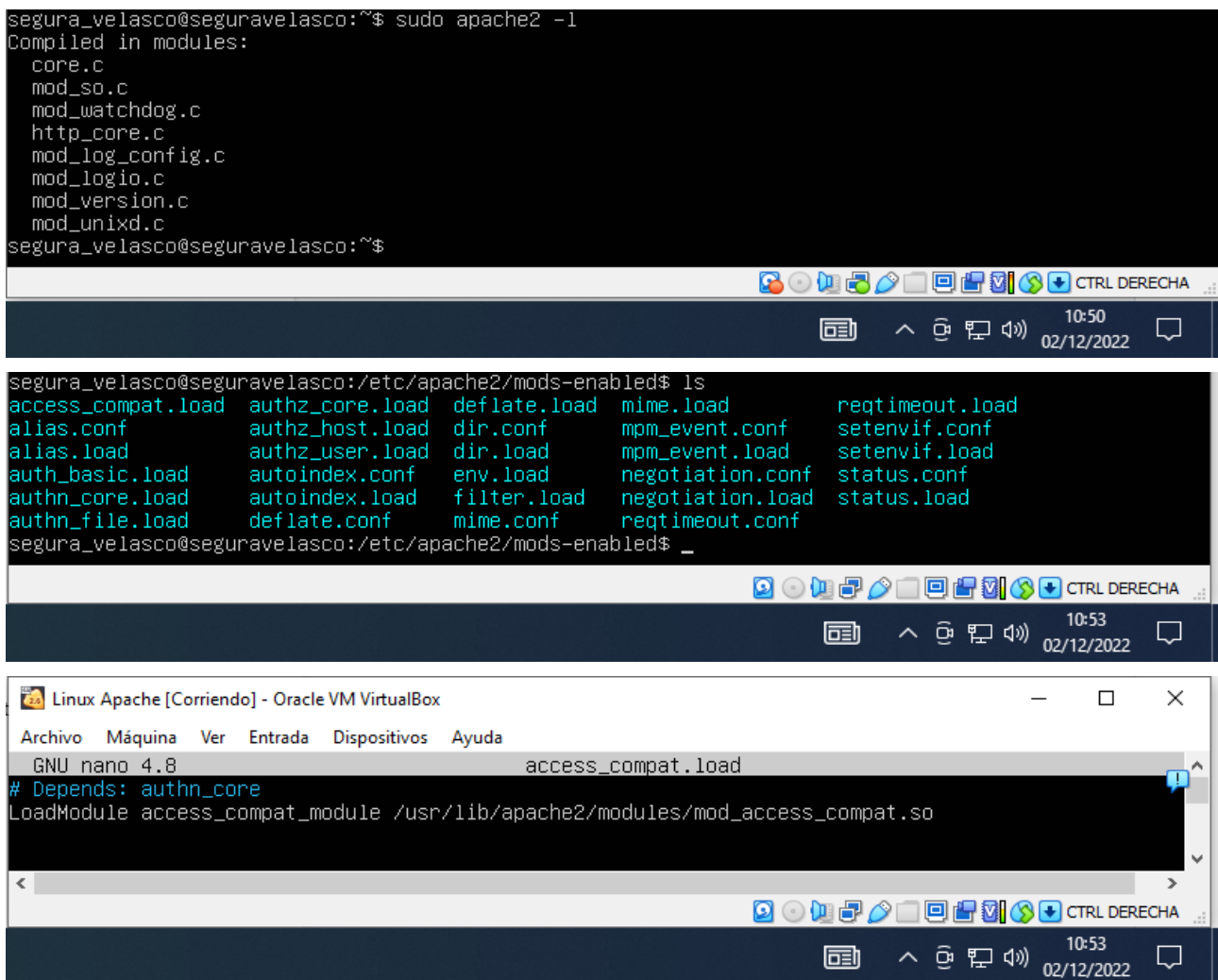
`/usr/lib/apache2/modules/mod_access_compat.so`

**PASO 4)** Edita uno de los archivos `.conf` y observa cómo se añaden directivas dentro del módulo. ¿Qué etiquetas se utilizan en estos archivos?

`AddType, AddOutputFilter, AddHandler, AddCharset, AddLanguage, RemoveType, TypesConfig...`

**PASO 5)** Consulta el directorio `/usr/lib/apache2/modules/` ¿qué archivos contiene? Un montón de archivos `"mod_x.so"` además de un único archivo `"httpd.exp"`

Toma capturas de los pasos 1, 2, 3 y 4.



The first screenshot shows a terminal window where the command `sudo apache2 -l` is executed. The output lists the modules compiled into the server: `core.c, mod_so.c, mod_watchdog.c, http_core.c, mod_log_config.c, mod_logio.c, mod_version.c, mod_unixd.c`.

The second screenshot shows a terminal window where the command `ls /etc/apache2/mods-enabled/` is executed. The output lists the enabled modules: `access_compat.load, auth_core.load, deflate.load, mime.load, reqtimeout.load, alias.conf, authz_core.load, dir.conf, mpm_event.conf, setenvif.conf, alias.load, authz_user.load, dir.load, mpm_event.load, setenvif.load, auth_basic.load, autoindex.conf, env.load, negotiation.conf, status.conf, authn_core.load, autoindex.load, filter.load, negotiation.load, status.load, authn_file.load, deflate.conf, mime.conf, reqtimeout.conf`.

The third screenshot shows a terminal window where the file `/etc/apache2/mods-enabled/access_compat.load` is being edited with the `nano` editor. The file content is as follows:

```
GNU nano 4.8 access_compat.load
# Depends: authn_core
LoadModule access_compat_module /usr/lib/apache2/modules/mod_access_compat.so
```

```
GNU nano 4.8      mime.conf
<IfModule mod_mime.c>

#
# TypesConfig points to the file containing the list of mappings from
# filename extension to MIME-type.
#
TypesConfig /etc/mime.types

#
# AddType allows you to add to or override the MIME configuration
# file mime.types for specific file types.
#
AddType application/x-gzip .tgz
#
# AddEncoding allows you to have certain browsers uncompress
# information on the fly. Note: Not all browsers support this.
# Despite the name similarity, the following Add* directives have
# nothing to do with the FancyIndexing customization directives above.
#
AddEncoding x-compress .Z
AddEncoding x-gzip .gz .tgz
AddEncoding x-bzip2 .bz2
#
# If the AddEncoding directives above are commented-out, then you
# probably should define those extensions to indicate media types:
#
AddType application/x-compress .Z
AddType application/x-gzip .gz .tgz
AddType application/x-bzip2 .bz2

#
# DefaultLanguage and AddLanguage allows you to specify the language of
# a document. You can then use content negotiation to give a browser a

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos      M-U Undo
^X Exit          ^R Read File    ^N Replace      ^U Paste Text   ^T To Spell     ^_ Go To Line    M-E Redo

10:58
02/12/2022
```

## A.2) Módulo userdir

El módulo **userdir** se utiliza para usar como directorio raíz del servidor HTTP el directorio home de un usuario.

Al utilizar este módulo, el usuario desde el que se va a usar, en el directorio raíz (/home/usuario) tendrá un directorio `public_html` que hará las veces de raíz web para Apache2.

En el caso de directorios raíz de usuarios, para acceder a ellos habrá que usar el carácter "~", o sea, la dirección será de la forma <http://hostname/~username/>

**PASO 1)** Comprueba si el módulo `userdir` está habilitado. ¿Lo está? No, no lo está

**PASO 2)** Si no lo está, habilita el módulo `userdir`.

**PASO 3)** Verifica ahora si el módulo está habilitado.

**PASO 4)** Reinicia el servidor para que los cambios tengan efecto.

**PASO 5)** Consulta el archivo `/etc/apache2/mods-enabled/userdir.conf`. ¿Cuál es el único usuario para el que está deshabilitado el uso de directorios personales? ¿Cuál es el subdirectorio que deben crear los usuarios en su carpeta home para poner sus páginas personales?

El usuario root está deshabilitado.

Podremos crear cualquier subdirectorio con el nombre que queramos, gracias al asterisco de la ruta del directorio.

**PASO 6)** Crea el directorio necesario dentro de tu usuario y añade un fichero denominado `personal.html` con el contenido Tu nombre e indicando que es personal.

**PASO 7)** Desde la máquina física, abre un navegador y accede al directorio raíz de tu usuario Linux.

### Index of /~profe

Name	Last modified	Size	Description
 <a href="#">Parent Directory</a>	-	-	-
 <a href="#">personal.html</a>	2016-11-22 19:55	38	

Apache/2.4.18 (Ubuntu) Server at 192.168.1.151 Port 80

**PASO 8)** Descarga el módulo y reinicia el servidor para que los cambios tengan efecto.

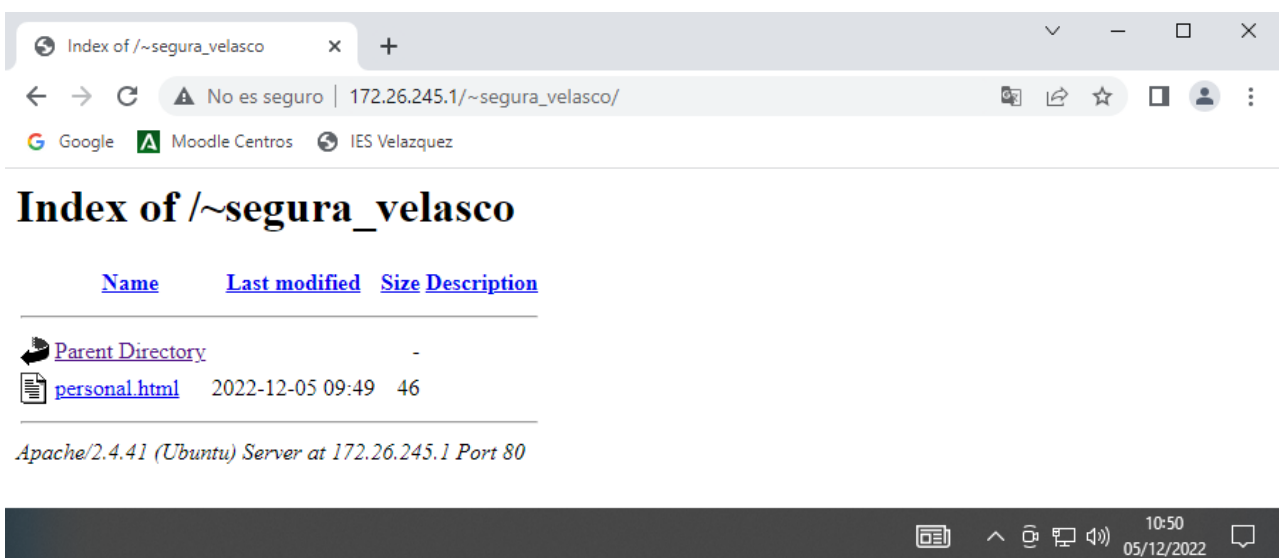
**Toma una captura de los pasos 3,5 y 7 (en esta última, donde se vea la barra de direcciones del navegador)**

```
segura_velasco@seguravelasco:/etc/apache2/mods-enabled$ sudo a2enmod userdir
Enabling module userdir.
To activate the new configuration, you need to run:
  systemctl restart apache2
segura_velasco@seguravelasco:/etc/apache2/mods-enabled$ systemctl restart apache2
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to restart 'apache2.service'.
Authenticating as: Daniel Segura (segura_velasco)
Password:
polkit-agent-helper-1: pam_authenticate failed: Authentication failure
==== AUTHENTICATION FAILED ====
Failed to restart apache2.service: Access denied
See system logs and 'systemctl status apache2.service' for details.
segura_velasco@seguravelasco:/etc/apache2/mods-enabled$ sudo systemctl restart apache2
segura_velasco@seguravelasco:/etc/apache2/mods-enabled$ ls
access_compat.load  authz_core.load  deflate.load  mime.load  reqtimeout.load  userdir.load
alias.conf          authz_host.load  dir.conf     mpm_event.conf  setenvif.conf
alias.load          authz_user.load  dir.load     mpm_event.load  setenvif.load
auth_basic.load     autoindex.conf  env.load     negotiation.conf status.conf
authn_core.load     autoindex.load  filter.load  negotiation.load status.load
authn_file.load     deflate.conf     mime.conf    reqtimeout.conf userdir.conf
segura_velasco@seguravelasco:/etc/apache2/mods-enabled$ _
```

```
GNU nano 4.8 userdir.conf
<IfModule mod_userdir.c>
  UserDir public_html
  UserDir disabled root

  <Directory /home/*/public_html>
    AllowOverride FileInfo AuthConfig Limit Indexes
    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
    Require method GET POST OPTIONS
  </Directory>
</IfModule>

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



### A.3) Módulo userdir en el servidor de clase

En el servidor del aula todos tenéis un usuario y una contraseña para entrar.

Recordad que es la inicial del primer nombre y el primer apellido.

Ejemplo: Amapola Gutiérrez de la Vega, sería agutierrez. La contraseña es alumno.

**PASO 1)** Accede al servidor a través de Putty. IP: 172.26.255.254

**PASO 2)** Da los pasos necesarios para qué al acceder a `http://172.26.255.254/~agutierrez` se vea tu página web en el servidor.

La página debe contener la IP de servidor y tu nombre completo



172.26.255.254/~linuxserver/

## Página WEB del usuario LINUXSERVER

### Detalla los pasos seguidos para conseguirlo.

1. Primero abrí PuTTY y entré en mi usuario "dsegura" usando la contraseña alumno
2. Creo dentro el directorio public\_html.
3. Creo la pagina html
4. La abro

```
172.26.255.254 - PuTTY
login as: dsegura
dsegura@172.26.255.254's password:
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.4.0-132-generic x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

System information as of lun 05 dic 2022 09:58:51 UTC

System load:  0.0           Temperature:   31.0 C
Usage of /:   17.5% of 54.22GB Processes:    137
Memory usage: 17%          Users logged in: 0
Swap usage:   0%           IPv4 address for enp2s0: 172.26.255.254

0 updates can be applied immediately.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

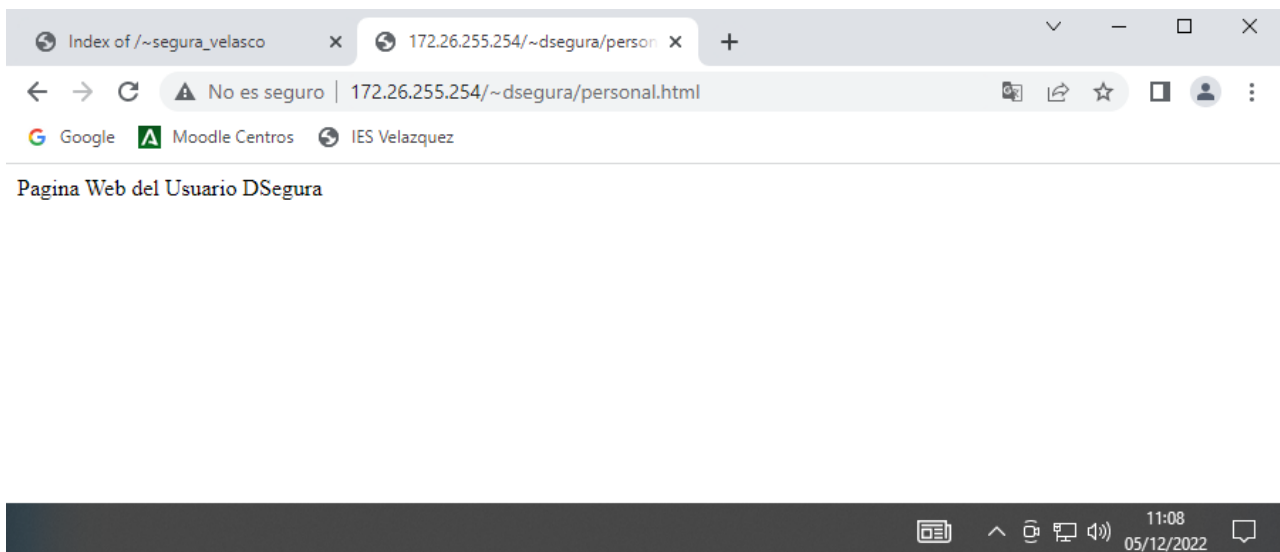
The programs included with the Ubuntu system are free software;
```



```
172.26.255.254 - PuTTY
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Thu Nov 25 08:28:38 2021 from 172.26.0.52
$ ls
$ mkdir public_html
$ nano /public_html/personal.html
$ ls
public_html
$ cd /public_html
-sh: 5: cd: can't cd to /public_html
$ sudo cd public_html
[sudo] password for dsegura:
dsegura is not in the sudoers file. This incident will be reported.
$ ls
public_html
$ cd public_html
$ ls
$ nano personal.html
$
```



## B) Control de acceso por IP y nombre de dominio

Para poder controlar el acceso a diferentes recursos dentro de nuestro servidor web podemos hacer uso del módulo **authz\_host**. Este módulo puede permitir o denegar el acceso a un recurso por parte de un host a partir de su dirección IP o su nombre de dominio.

Más información del módulo en: [https://httpd.apache.org/docs/2.4/mod/mod\\_authz\\_host.html](https://httpd.apache.org/docs/2.4/mod/mod_authz_host.html)

Vamos a controlar el acceso a un recurso de Apache en nuestro servidor Linux para que la máquina física tenga acceso, y la máquina de un compañero no:

**PASO 1)** Comprueba si está habilitado el módulo **authz\_host**. ¿Lo está? Lo está

```
segura_velasco@seguravelasco:~$ cd /etc/apache2/mods-enabled
segura_velasco@seguravelasco:/etc/apache2/mods-enabled$ ls
access_compat.load  authz_core.load  deflate.load  mime.load  reqtimeout.load  userdir.load
alias.conf          authz_host.load  dir.conf     mpm_event.conf  setenvif.conf
alias.load          authz_user.load  dir.load     mpm_event.load  setenvif.load
auth_basic.load     autoindex.conf  env.load     negotiation.conf status.conf
authn_core.load     autoindex.load  filter.load  negotiation.load status.load
authn_file.load     deflate.conf     mime.conf    reqtimeout.conf userdir.conf
segura_velasco@seguravelasco:/etc/apache2/mods-enabled$ _
```

**PASO 2)** Crea un directorio **/var/www/html/tuNombre/**. Dentro del directorio crea un archivo y llámalo **tuNombre.html** y añade el contenido que quieras.

```
segura_velasco@seguravelasco:/var/www/html$ ls
ciclos  despliegue.html  Dsegura  fp.html  indice.html
segura_velasco@seguravelasco:/var/www/html$ cd /var/www/html/Dsegura
segura_velasco@seguravelasco:/var/www/html/Dsegura$ ls
Dsegura.html
segura_velasco@seguravelasco:/var/www/html/Dsegura$
```

**PASO 2)** Edita el fichero de configuración **/etc/apache2/sites-available/000-default.conf** y añade la directiva **Directory** para el recurso creado anteriormente.

**PASO 3)** Añade dentro de la directiva anterior las directivas de acceso necesarias para que la máquina física, a partir de su dirección IP, pueda acceder a este recurso pero no la máquina del compañero (échale un vistazo al enlace informativo del módulo **authz\_host** que hay más arriba).

A la hora de hacer este paso, la IP de nuestra máquina física es 172.26.0.46

**PASO 4)** Reinicia el servidor para que los cambios tengan efecto.

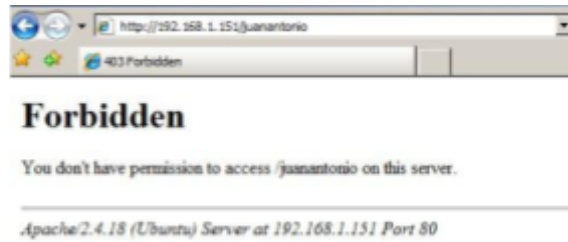
**PASO 5)** Abre un navegador desde tu máquina física e intenta acceder al recurso **/tuNombre/** y comprueba que se puede.

**PASO 6)** Abre un navegador desde la máquina del compañero e intenta acceder al recurso **/tuNombre/** y comprueba que no se puede.

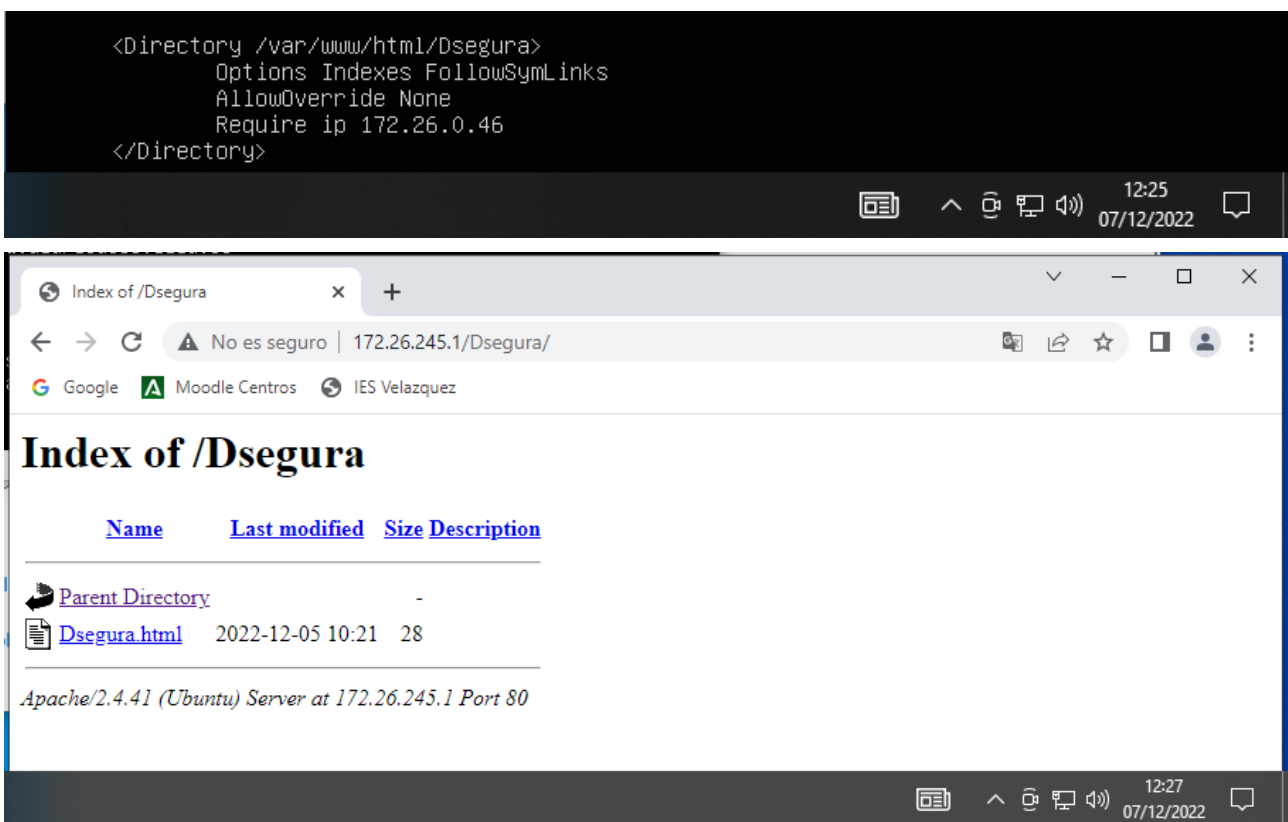
Desde mi máquina física:

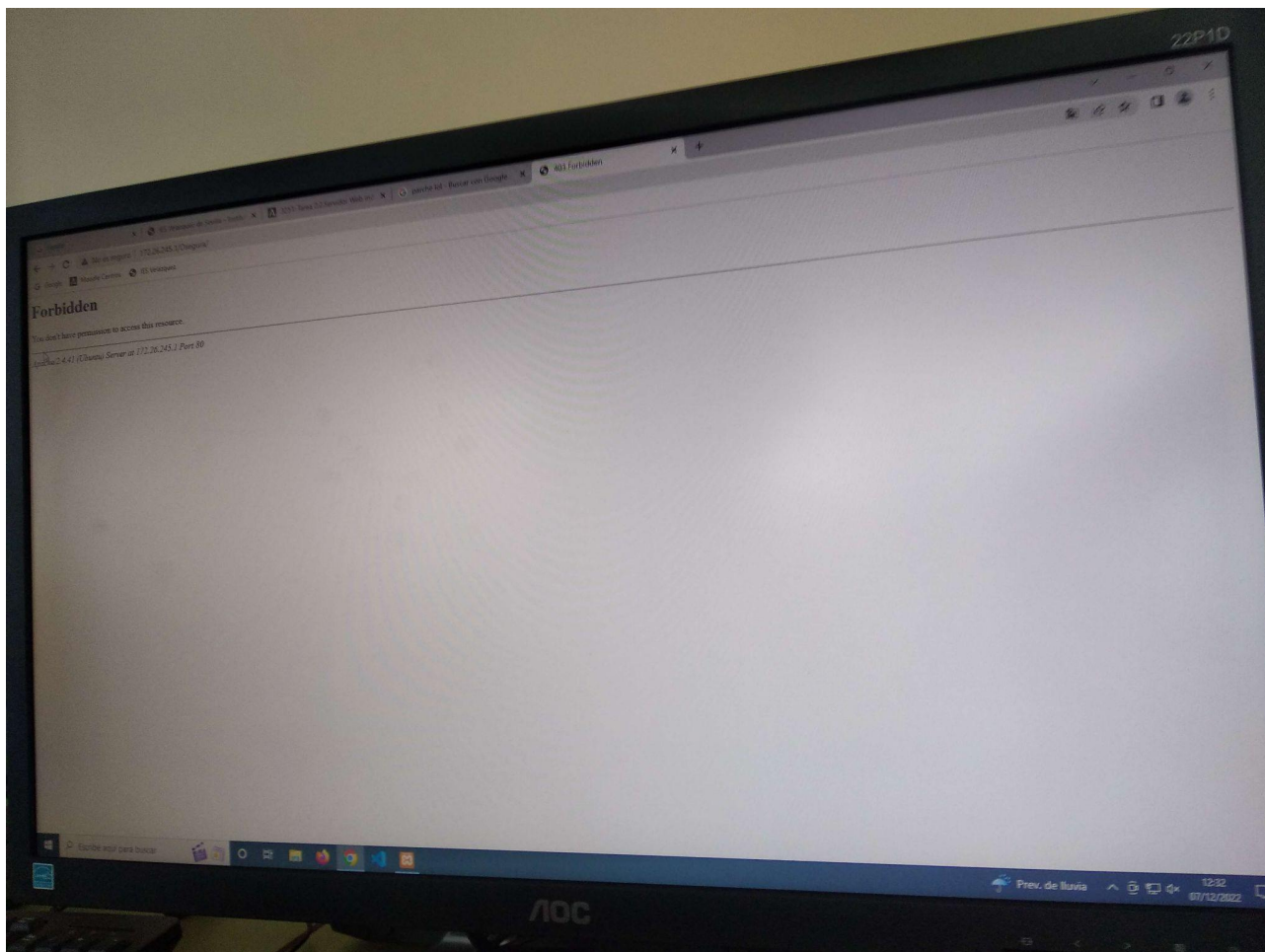


Desde la máquina del compañero:



Toma una captura de los pasos 3,4,5 y 6.



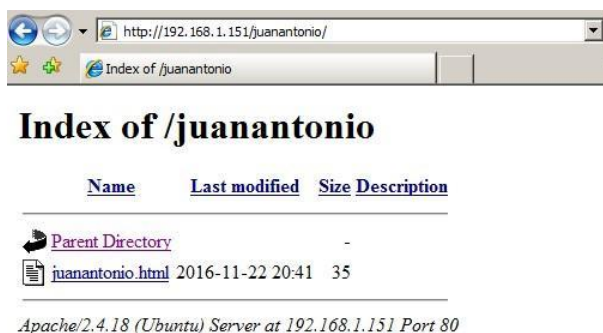


**PASO 7)** Añade el acceso al recurso de tu carpeta para la máquina del compañero pero **usando su nombre de host en vez de su IP**.

**PASO 8)** Reinicia el servidor para que los cambios tengan efecto.

**PASO 9)** Abre un navegador desde la máquina del compañero e intenta acceder al recurso **/tuNombre/** y comprueba que ahora sí se puede. Me sigue sin poder meter en la carpeta, estando aún Forbidden.

Desde la máquina del compañero:



Toma una captura de los pasos 7 y 9.

```
<Directory /var/www/html/Dsegura>  
  Options Indexes FollowSymLinks  
  AllowOverride None  
  Require host AULA43-PC07  
</Directory>
```

12:38  
07/12/2022

### C) Autenticación y autorización Basic y Digest

La autenticación es el proceso mediante el cual se puede verificar que alguien es quien dice ser. La autorización es el proceso mediante el cual se permite a acceder a un recurso solicitado.

En este punto vamos a usar las autenticaciones Basic y Digest.

(<http://httpd.apache.org/docs/2.2/es/howto/auth.html>)

Autenticación Basic:

- La contraseña es enviada por el cliente en texto plano.
- Autenticación y autorización sobre fichero de texto (comando **htpasswd**).
- Usa los módulos **authn\_file** y **authz\_user**.

```
# La primera vez que se invoca el comando se
# utiliza la opción -c para crear el fichero
htpasswd -c /etc/apache2/passwd profesor1

# Añade un nuevo usuario al fichero
htpasswd /etc/apache2/passwd profesor2

# Borrar un nuevo usuario al fichero
htpasswd -D /etc/apache2/passwd profesor1
```

<http://httpd.apache.org/docs/2.2/es/programs/htpasswd.html>

- Definir directivas:
  - **AuthType**: tipo de autorización
  - **AuthName**: nombre de la autorización cuando el cliente reciba el mensaje
  - **AuthUserFile**: localización del fichero donde están los usuarios que pueden autenticarse
  - **Require**: solo los usuarios o grupos de usuarios que aparecen en esta directiva pueden acceder al recurso.

```
<Directory /var/www/profesor>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from 127.0.0.1
    allow from 192.168.1.16
    AuthType Basic
    AuthName "Acceso restringido"
    AuthUserFile /etc/apache2/passwd
    Require user profesor1 profesor2
</Directory>
```

Autenticación digest:

- La contraseña se envía cifrada (cifrado débil) por el cliente.
- Autenticación y autorización sobre fichero de texto (comando **htdigest**)
- Módulos: **mod\_auth\_digest** y **mod\_auth\_user**



```
# La primera vez que se invoca el comando se
# utiliza a opción -c para crear el fichero
htdigest -c /etc/apache2/digest    informatica admin1

# Añade un nuevo usuario al fichero
Htdigest /etc/apache2/digest    informatica admin2

# Borrar un nuevo usuario al fichero
htdigest -D /etc/apache2/digest    informatica admin1
```

<http://httpd.apache.org/docs/2.2/es/programs/htdigest.html>

- Definir directivas:
  - AuthType: tipo de autorización
  - AuthName: nombre de la autorización cuando el cliente reciba el mensaje
  - AuthDigestProvider: establecen el método de almacenamiento de las contraseñas del servidor, en nuestro caso se almacenarán en un archivo y por tanto tendrán el valor file
  - AuthUserFile: localización del fichero donde están los usuarios que pueden autenticarse
  - Require solo los usuarios o grupos de usuarios que aparecen en esta directiva pueden acceder al recurso

```
<Directory /var/www/departamento>
Options Indexes FollowSymLinks MultiViews
AllowOverride None
AuthType Digest
AuthName "informatica"
AuthDigestProvider file
AuthUserFile /etc/apache2/digest
Require user admin1 admin2
</Directory>
```

En este punto vamos a configurar la autenticación Basic y Digest para recursos de Apache en nuestro servidor Linux.

### C.1) Autenticación Basic

**PASO 1)** Comprueba si el módulo **auth\_basic** está habilitado, si no lo está, habilítalo. Lo tenemos habilitado.

**PASO 2)** Vamos a crear el directorio **/nombreAlumno/** dentro de nuestro directorio raíz **/var/www/html/**. Dentro añadiremos un archivo **nombreAlumno.html** donde incluiremos el contenido que queramos.

**PASO 3)** Para usar la autenticación Basic hay que crear un fichero accesible (el fichero que se creará será **/etc/apache2/passwd**) en el que se guardarán los usuarios y contraseñas. Para crear ese fichero se utilizará el comando **htpasswd** (ver cuadro arriba). Añade los usuarios **apellido1** y **apellido2**.

La contraseña utilizada para ambos usuarios será "daniel".

**PASO 4)** Edita el fichero de configuración **/etc/apache2/sites-available/000-default.conf** y permite el acceso al directorio **/var/www/html/nombreAlumno** a los usuarios **apellido1** y **apellido2** (ver cuadro ejemplo arriba).

**PASO 5)** Reinicia el servidor para que los cambios tengan efecto.

**PASO 6)** Abre un navegador desde tu máquina física y accede al recurso `/nombreAlumno` como usuario `apellido1`.

**PASO 7)** Abre un navegador desde la máquina de un compañero y accede al recurso `/nombreAlumno` como usuario `apellido2`.

**Toma capturas de los pasos 3,4, 6 y 7 (de estas últimos una captura cuando sale el cuadro para autenticarte y luego una vez dentro del recurso `/amigo`).**

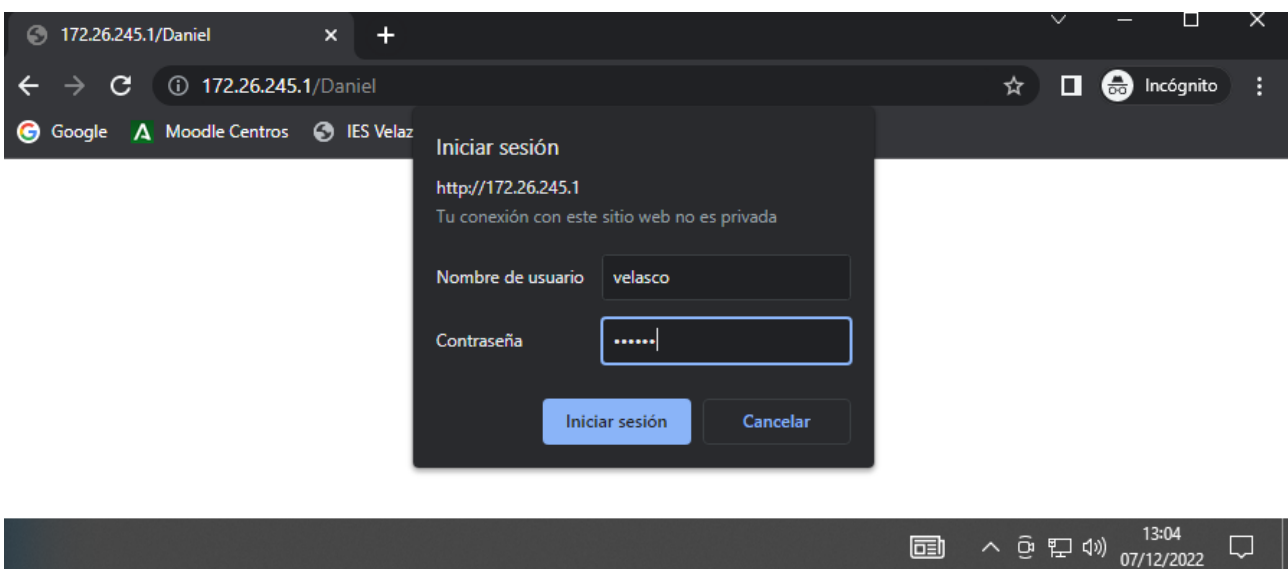
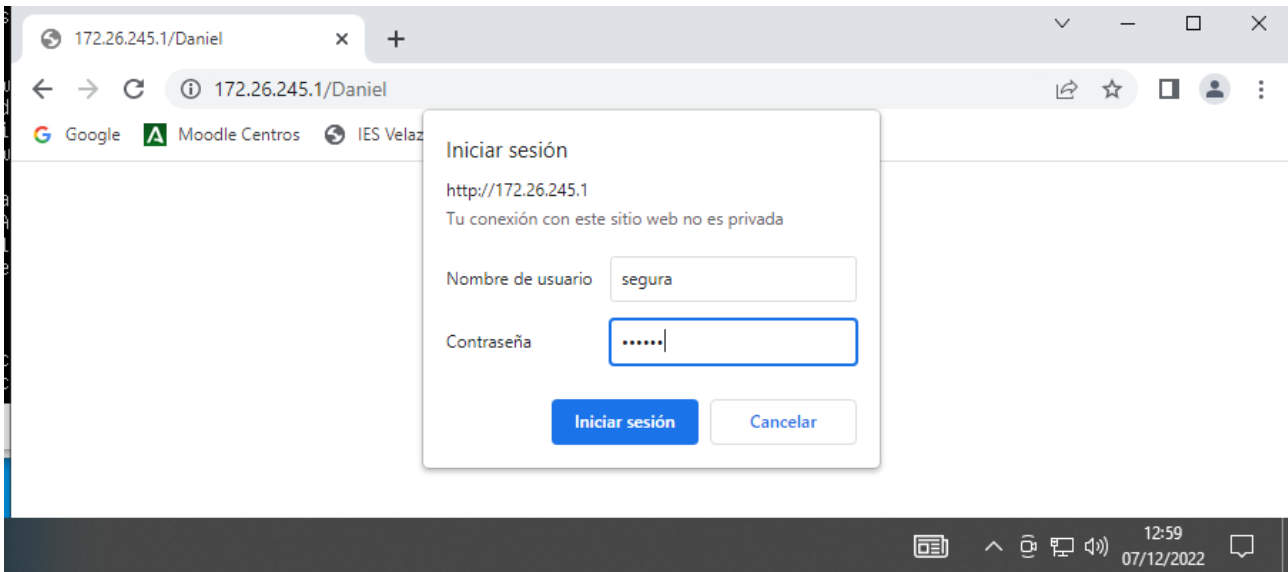
```
segura_velasco@seguravelasco:~$ sudo mkdir /var/www/html/Daniel
segura_velasco@seguravelasco:~$ sudo nano /var/www/html/Daniel/Daniel.html
```

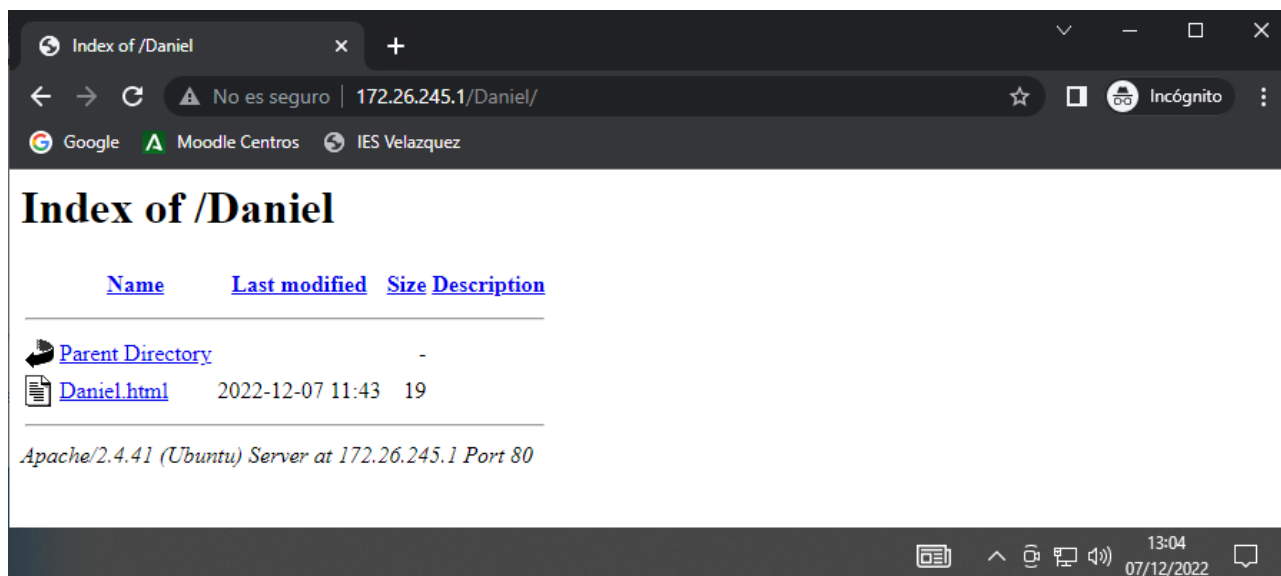
```
segura_velasco@seguravelasco:/etc/apache2$ sudo htpasswd -c /etc/apache2/passwd segura
New password:
Re-type new password:
Adding password for user segura
segura_velasco@seguravelasco:/etc/apache2$ cat passwd
segura:$apr1$GcUdG0Am$X1Z0K1tw62r.RQbcg0/An0
segura_velasco@seguravelasco:/etc/apache2$ sudo htpasswd /etc/apache2/passwd segura
New password:
Re-type new password:
htpasswd: password verification error
segura_velasco@seguravelasco:/etc/apache2$ sudo htpasswd /etc/apache2/passwd velasco
New password:
Re-type new password:
Adding password for user velasco
segura_velasco@seguravelasco:/etc/apache2$ cat passwd
segura:$apr1$GcUdG0Am$X1Z0K1tw62r.RQbcg0/An0
velasco:$apr1$ussbz6D0$N5zQakFUpyHxBaHihGQ21
segura_velasco@seguravelasco:/etc/apache2$ _
```

```
<Directory /var/www/html/Daniel>
    Options Indexes FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from 172.26.0.46
    AuthType Basic
    AuthName "Acceso Restringido"
    AuthUserFile /etc/apache2/passwd
    Require user_segura velasco
</Directory>
```

```
^G Get Help    ^O Write Out  ^W Where Is   ^K Cut Text    ^J Justify    ^C Cur Pos    M-U Undo
^X Exit        ^R Read File  ^N Replace    ^U Paste Text  ^T To Spell   ^G Go To Line M-E Redo
```







## C.2) Autenticación Digest

**PASO 1)** Comprueba si el módulo **auth\_digest** está habilitado, si no lo está, habilítalo.

No lo tenemos habilitado, así que toca poner "sudo a2enmod auth\_digest" y reiniciamos.

**PASO 2)** Vamos a crear el directorio **/tareac2/** dentro de nuestro directorio raíz **/var/www/html/**. Dentro añadiremos un archivo **tareac2.html** donde incluiremos el contenido que queramos.

**PASO 3)** Para usar la autenticación **Digest** también hay que crear un fichero accesible (el fichero que se creará será también **/etc/apache2/passwd** pero para **digest**) en el que se guardarán los usuarios y contraseñas, pero esta vez asociados a un dominio (en el cuadro ejemplo de arriba el dominio o "realm" es informática). Para crear ese fichero se utilizará el comando **httdigest** (ver cuadro arriba). Añade los usuarios **inicialPrimerApellidoNombre** y **inicialSegundoApellidoNombre**.

Daniel Segura Velasco

sdaniel

vdaniel

Contraseña: daniel

**PASO 4)** Edita el fichero de configuración **/etc/apache2/sites-available/000-default.conf** y permite el acceso al directorio **/var/www/html/tareac2** a los usuarios **inicialPrimerApellidoNombre** y **inicialSegundoApellidoNombre** (ver cuadro ejemplo arriba). Ten en cuenta que en la directiva **AuthName** tienes que poner lo mismo que pusiste en el dominio o "realm".

**PASO 5)** Reinicia el servidor para que los cambios tengan efecto.

**PASO 6)** Abre un navegador desde tu máquina física y accede al recurso **/tareac2** como usuario **inicialPrimerApellidoNombre**.

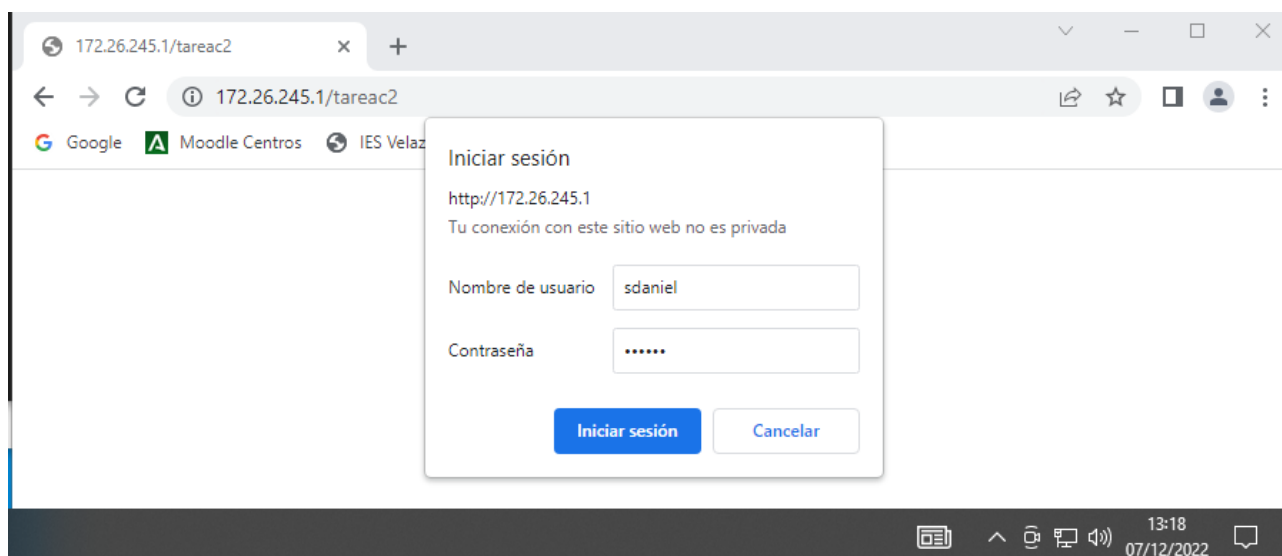
**PASO 7)** Abre un navegador desde la máquina de un compañero y accede al recurso **/tareac2** como usuario **inicialSegundoApellidoNombre**.

**Toma una captura de los pasos 3, 4, 6 y 7 (de estas últimos una captura cuando sale el cuadro para autenticarte y luego una vez dentro del recurso /primo).**

```
segura_velasco@seguravelasco:~$ sudo htdigest /etc/apache2/passwd informatica sdaniel
Adding user sdaniel in realm informatica
New password:
Re-type new password:
segura_velasco@seguravelasco:~$ sudo htdigest /etc/apache2/passwd informatica vdaniel
Adding user vdaniel in realm informatica
New password:
Re-type new password:
segura_velasco@seguravelasco:~$ cd /etc/apache2
segura_velasco@seguravelasco:/etc/apache2$ cat passwd
segura:$apr1$GcUdG0Am$Xl20K1tw62r.RQbcg0/An0
velasco:$apr1$ussbz6DD$N5zQakFUpYHx8AHiihGQ21
sdaniel:informatica:2bb55608a15559792ad8581cb91bcc8b
vdaniel:informatica:fae3a6f9a12988f71203aae30a6a8a16
segura_velasco@seguravelasco:/etc/apache2$ _
```

```
<Directory /var/www/html/tareac2>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    AuthType Digest
    AuthName "informatica"
    AuthDigestProvider file
    AuthUserFile /etc/apache2/passwd
    Require user sdaniel vdaniel
</Directory>

<Directory /var/www/html/ciclos>
    DirectoryIndex daw.html
```



Index of /tareac2

No es seguro | 172.26.245.1/tareac2/

Google Moodle Centros IES Velazquez

## Index of /tareac2

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>	-		
<a href="#">tareac2.html</a>	2022-12-07 12:10	12	

Apache/2.4.41 (Ubuntu) Server at 172.26.245.1 Port 80

13:19 07/12/2022

172.26.245.1/tareac2

172.26.245.1/tareac2

Google Moodle Centros IES Velazquez

Incógnito

Iniciar sesión

http://172.26.245.1

Tu conexión con este sitio web no es privada

Nombre de usuario

Contraseña

Iniciar sesión Cancelar

13:20 07/12/2022

Index of /tareac2

No es seguro | 172.26.245.1/tareac2/

Google Moodle Centros IES Velazquez

## Index of /tareac2

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>	-		
<a href="#">tareac2.html</a>	2022-12-07 12:10	12	

Apache/2.4.41 (Ubuntu) Server at 172.26.245.1 Port 80

13:21 07/12/2022

#### **D) Ficheros .htaccess (si no sale poner pantallazo de haberlo intentado)**

Los archivos `.htaccess` permiten configurar de manera personalizada directorios concretos que se quieran servir desde el Servidor Apache, pero sin que estos cambios afecten a la configuración general del servidor Apache. Básicamente permite "personalizar" el cómo se sirven unos contenidos que pertenecen a un directorio concreto.

Para poder hacer uso de los ficheros `.htaccess` tenemos que permitir en el archivo de configuración de apache (`httpd.conf`) su uso mediante la directiva `AllowOverride`.

**PASO 1)** Crea el usuario `useraccess`.

Su contraseña es "daniel".

**PASO 2)** Abre el fichero de configuración `000-default` y crea el **alias** `myBlog` dentro de la carpeta personal del nuevo usuario `useraccess`. Deja como única directiva `AllowOverride All`.

```
Alias /myBlog /home/useraccess/myBlog
<Directory /home/useraccess/myBlog>
    AllowOverride All
</Directory>
```

**PASO 3)** Reinicia el servidor para que los cambios tengan efecto.

**PASO 4)** Inicia sesión con el nuevo usuario `useraccess`.

**PASO 5)** Crea dentro del directorio `home` de este usuario el **directorio** `myBlog`. Crea dentro el archivo `myBlog.html` con el contenido que quieras.

**PASO 6)** Para el acceso a los recursos de `myBlog` vamos a usar un tipo de autenticación `Digest`, por lo que dentro de este directorio vamos a crear el fichero `.htdigest` para el servidor informática y para el usuario `myUserBlog` (ver punto anterior acceso mediante `Digest`).

**PASO 7)** Ahora tendremos que crear el fichero `.htaccess` (también dentro de `myBlog`).


Dentro añadiremos las directivas necesarias para que se acceda solo desde nuestra máquina física (no es necesario poner las directivas `Directory` pues ya las incluimos en nuestro `Alias` para este directorio dentro de `000-default`).

```
Options Indexes
Order allow,deny
allow from 192.168.1.101
AuthType Digest
AuthName "informatica"
AuthUserFile /home/useraccess/myBlog/.htdigest
Require user myUserBlog
```

**PASO 8)** Vamos a acceder desde nuestra máquina física al recurso **myBlog** para ver que nos pide la autenticación y que podemos acceder al recurso.



**Toma una captura de los pasos 2,6,7 y 8.**

 Linux Apache [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

```
segura_velasco@seguravelasco:~$ sudo adduser useraccess
Adding user `useraccess' ...
Adding new group `useraccess' (1001) ...
Adding new user `useraccess' (1001) with group `useraccess' ...
Creating home directory `/home/useraccess' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for useraccess
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
segura_velasco@seguravelasco:~$ _
```

```
Alias /myBlog /home/useraccess/myBlog
<Directory /home/useraccess/myBlog>
    AllowOverride All
</Directory>
```

```
useraccess@seguravelasco:~/myBlog$ sudo htdigest -c /home/useraccess/myBlog/.htdigest informatica myUserBlog
[sudo] password for useraccess:
useraccess is not in the sudoers file. This incident will be reported.
useraccess@seguravelasco:~/myBlog$ htdigest -c /home/useraccess/myBlog/.htdigest informatica myUserBlog
Adding password for myUserBlog in realm informatica.
New password:
Re-type new password:
useraccess@seguravelasco:~/myBlog$
```

Linux Apache [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

GNU nano 4.8 /home/useraccess/myBlog/.htaccess

```
Options Indexes
Order allow,deny
allow from 172.26.0.46
AuthType Digest
AuthName "informatica"
AuthUserFile /home/useraccess/myBlog/.htdigest
Require user myUserBlog
```

172.26.245.1/myBlog

172.26.245.1/myBlog

Google Moodle Centros IES Velaz

Iniciar sesión

http://172.26.245.1

Tu conexión con este sitio web no es privada

Nombre de usuario



Contraseña

Index of /myBlog

No es seguro | 172.26.245.1/myBlog/

Google Moodle Centros IES Velazquez

## Index of /myBlog

Name	Last modified	Size	Description
 <a href="#">Parent Directory</a>		-	
 <a href="#">myBlog.html</a>	2022-12-07 12:34	13	

Apache/2.4.41 (Ubuntu) Server at 172.26.245.1 Port 80

13:44 07/12/2022



## E) Ficheros de registros (logs)

Los ficheros de registros nos ofrecen información de errores y accesos del servidor Apache.

En linux los ficheros de registro son:

Errores **/var/log/apache2/error.log**

Accesos **/var/log/apache2/access.log**

En windows:

Error **C:\Program Files\Apache Software Foundation\Apache2.2\log\error.log**

Accesos **C:\Program Files\Apache Software Foundation\Apache2.2\log\access.log**

Algunas de las directivas que tienen que ver con estos ficheros de registros son:

ErrorLog: Especifica los archivos donde se guardan los errores del servidor

LogLevel: Establece el nivel de detalle de los registros de mensajes de error

CustomLog: Identifica el archivo de registro de accesos y su formato (por defecto, combined)

LogFormat: Configura el formato para los archivos de registros del servidor Web (realmente depende de la configuración dada en CustomLog).

**PASO 1)** En tu servidor Linux, consulta el fichero 000-default y responde a las siguientes preguntas:

- ¿Qué directiva marca la ruta del archivo de los errores? ¿Cuál es el fichero de logs de errores? ¿Qué nivel de prioridad tiene?

La directiva ErrorLog marca la ruta de errores.

El fichero de los errores es: `"/var/log/apache2/error.log"`.

Y el nivel de prioridad, que es igual al LogLevel, viene por defecto en `"warn"`.

- ¿Qué directiva marca la ruta del archivo de los accesos? ¿Cuál es el fichero de logs de accesos?

La directiva CustomLog marca la ruta de accesos

El fichero al que van los accesos es `"/var/log/apache2/access.log"`.

**PASO 2)** Consulta el log de errores

**PASO 3)** Consulta el log de accesos

Toma una captura de los pasos 2 y 3 (del final de cada fichero).

```
[Wed Dec 07 12:04:13.556503 2022] [authz_host:error] [pid 1805:tid 139774375474944] [client 172.26.0.46]
[Wed Dec 07 12:04:13.556534 2022] [authz_core:error] [pid 1805:tid 139774375474944] [client 172.26.0.46]
[Wed Dec 07 12:09:28.836157 2022] [mpm_event:notice] [pid 1803:tid 139774622039104] AH00491: caught
[Wed Dec 07 12:09:28.896006 2022] [mpm_event:notice] [pid 1929:tid 140216756419648] AH00489: Apache
[Wed Dec 07 12:09:28.896084 2022] [core:notice] [pid 1929:tid 140216756419648] AH00094: Command line
[Wed Dec 07 12:17:14.677797 2022] [mpm_event:notice] [pid 1929:tid 140216756419648] AH00491: caught
[Wed Dec 07 12:17:14.717876 2022] [mpm_event:notice] [pid 2038:tid 139832325594176] AH00489: Apache
[Wed Dec 07 12:17:14.717946 2022] [core:notice] [pid 2038:tid 139832325594176] AH00094: Command line
[Wed Dec 07 12:17:41.155610 2022] [auth_digest:error] [pid 2040:tid 139832189777664] [client 172.26.0.46]
[Wed Dec 07 12:20:06.141709 2022] [authz_host:error] [pid 2041:tid 139832172992256] [client 172.26.0.46]
[Wed Dec 07 12:20:06.141787 2022] [authz_core:error] [pid 2041:tid 139832172992256] [client 172.26.0.46]
[Wed Dec 07 12:29:53.798654 2022] [mpm_event:notice] [pid 2038:tid 139832325594176] AH00491: caught
[Wed Dec 07 12:29:53.837651 2022] [mpm_event:notice] [pid 2166:tid 140343273622592] AH00489: Apache
[Wed Dec 07 12:29:53.837719 2022] [core:notice] [pid 2166:tid 140343273622592] AH00094: Command line
[Wed Dec 07 12:43:37.544884 2022] [auth_digest:error] [pid 2168:tid 140343262729984] [client 172.26.0.46]
[ File '/var/log/apache2/error.log' is unwritable ]
^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify    ^C Cur Pos    M-U Undo
^X Exit          ^R Read File   ^N Replace    ^U Paste Text  ^T To Spell   ^_ Go To Line  M-E Redo

13:50
07/12/2022

172.26.0.46 - - [07/Dec/2022:11:58:12 +0000] "GET /Daniel HTTP/1.1" 401 731 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
172.26.0.46 - - [07/Dec/2022:11:59:04 +0000] "-" 408 0 "-" "-"
172.26.0.46 - segura [07/Dec/2022:11:59:30 +0000] "GET /Daniel HTTP/1.1" 401 731 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
172.26.0.46 - segura [07/Dec/2022:11:59:45 +0000] "GET /Daniel HTTP/1.1" 401 731 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
172.26.0.46 - - [07/Dec/2022:12:00:36 +0000] "-" 408 0 "-" "-"
172.26.0.46 - segura [07/Dec/2022:12:03:05 +0000] "GET /Daniel HTTP/1.1" 301 577 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
172.26.0.46 - segura [07/Dec/2022:12:03:05 +0000] "GET /Daniel/ HTTP/1.1" 200 705 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
172.26.0.46 - - [07/Dec/2022:12:03:57 +0000] "-" 408 0 "-" "-"
aula43-pc09 - - [07/Dec/2022:12:04:13 +0000] "GET / HTTP/1.1" 200 735 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
aula43-pc09 - - [07/Dec/2022:12:04:13 +0000] "GET /icons/blank.gif HTTP/1.1" 200 431 "http://172.26.0.46/"
aula43-pc09 - - [07/Dec/2022:12:04:13 +0000] "GET /icons/folder.gif HTTP/1.1" 200 508 "http://172.26.0.46/"
aula43-pc09 - - [07/Dec/2022:12:04:13 +0000] "GET /icons/text.gif HTTP/1.1" 200 512 "http://172.26.0.46/"
172.26.0.46 - - [07/Dec/2022:12:04:20 +0000] "GET /Daniel HTTP/1.1" 401 731 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
172.26.0.46 - velasco [07/Dec/2022:12:04:56 +0000] "GET /Daniel HTTP/1.1" 301 577 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
172.26.0.46 - velasco [07/Dec/2022:12:04:56 +0000] "GET /Daniel/ HTTP/1.1" 200 705 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
172.26.0.46 - - [07/Dec/2022:12:04:56 +0000] "GET /icons/back.gif HTTP/1.1" 200 499 "http://172.26.0.46/"
172.26.0.46 - - [07/Dec/2022:12:17:41 +0000] "GET /tareac2 HTTP/1.1" 401 814 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0"
[ File '/var/log/apache2/access.log' is unwritable ]
^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify    ^C Cur Pos    M-U Undo
^X Exit          ^R Read File   ^N Replace    ^U Paste Text  ^T To Spell   ^_ Go To Line  M-E Redo

13:51
07/12/2022
```

## F) Módulos status e info

`status` e `info` son módulos de monitorización. En concreto:

`status` permite monitorizar el rendimiento del servidor Apache (generando un HTML).

`info` proporciona una vista resumida de la configuración del servidor.

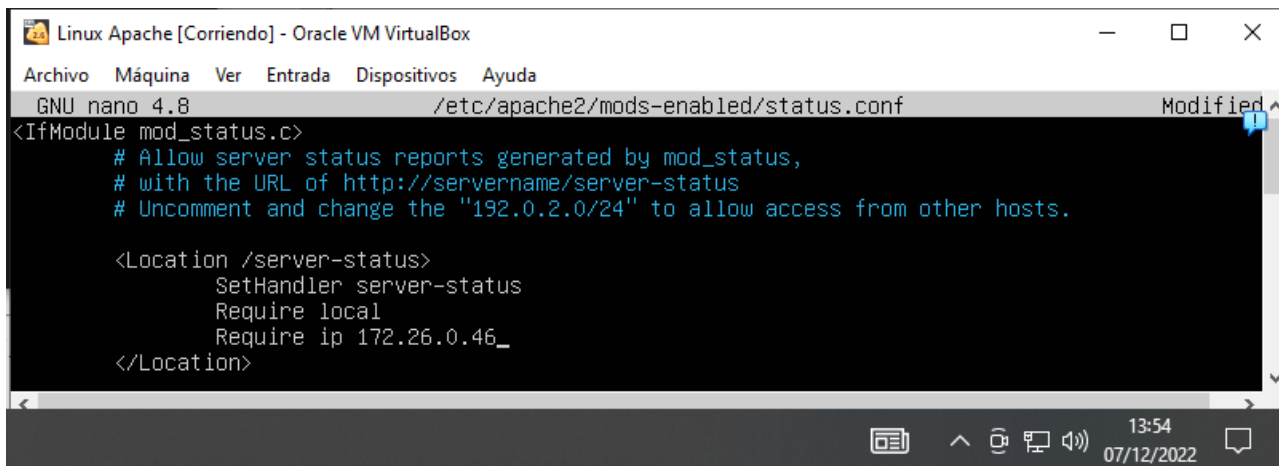
**PASO 1)** En tu servidor Linux, habilita el módulo **status**. Esta habilitado

**PASO 2)** El fichero de configuración del módulo es **status.conf**, edita el fichero y habilita el acceso desde tu máquina física.

**PASO 3)** Reinicia el servidor para aplicar los cambios.

**PASO 4)** Desde tu máquina física conéctate al recurso `server-status`

**Toma una captura de los pasos 2 y 4.**



```
Linux Apache [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 4.8 /etc/apache2/mods-enabled/status.conf Modified
<IfModule mod_status.c>
# 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 other hosts.

<Location /server-status>
    SetHandler server-status
    Require local
    Require ip 172.26.0.46_
</Location>
```

Apache Status

No es seguro | 172.26.245.1/server-status

Google Moodle Centros IES Velazquez

## Apache Server Status for 172.26.245.1 (via 172.26.245.1)

Server Version: Apache/2.4.41 (Ubuntu)  
Server MPM: event  
Server Built: 2022-06-14T13:30:55

---

Current Time: Wednesday, 07-Dec-2022 12:58:08 UTC  
Restart Time: Wednesday, 07-Dec-2022 12:57:49 UTC  
Parent Server Config. Generation: 1  
Parent Server MPM Generation: 0  
Server uptime: 18 seconds  
Server load: 0.00 0.00 0.00  
Total accesses: 0 - Total Traffic: 0 kB - Total Duration: 0  
CPU Usage: u0 s0 cu0 cs0  
0 requests/sec - 0 B/second  
1 requests currently being processed, 49 idle workers

Slot	PID	Stopping	Connections		Threads		Async connections		
			total	accepting	busy	idle	writing	keep-alive	closing
0	2320	no	0	yes	1	24	0	0	0
1	2321	no	0	yes	0	25	0	0	0
Sum	2	0	0		1	49	0	0	0

W .....

Scoreboard Key:  
"\_" Waiting for Connection, "s" Starting up, "r" Reading Request,  
"w" Sending Reply, "k" Keepalive (read), "D" DNS Lookup,  
"c" Closing connection, "l" Logging, "G" Gracefully finishing,  
"I" Idle cleanup of worker, "." Open slot with no current process

Srv	PID	Acc	M	CPU	SS	Req	Dur	Conn	Child	Slot	Client	Protocol	VHost	Request
0-0	2320	1/0/0	W	0.00	0	0	0	0.0	0.00	0.00	172.26.0.46	http/1.1	172.26.245.1:80	GET /server-status HTTP/1.1

13:58  
07/12/2022

**PASO 5)** En tu servidor Linux, habilita el módulo **info**.

**PASO 6)** El fichero de configuración del módulo es **info.conf**, edita el fichero y habilita el acceso desde tu máquina física.

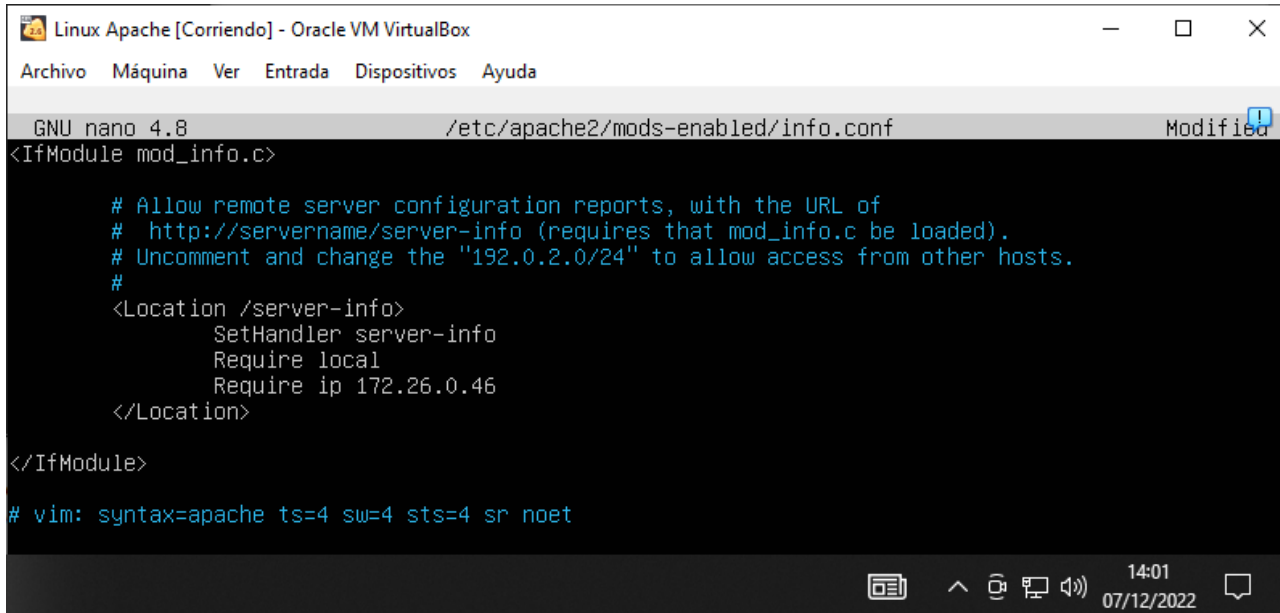
**PASO 7)** Reinicia el servidor para aplicar los cambios.

**PASO 8)** Desde tu máquina física conéctate al recurso server-info

Consulta el fichero server-info, ¿tienes cargado el módulo mod\_mime? ¿en caso que lo tuvieras, tiene el módulo cargada la configuración de caracteres UTF-32?

Toma una captura de los pasos 6 y 8.

```
segura_velasco@seguravelasco:~$ sudo a2enmod info
Enabling module info.
To activate the new configuration, you need to run:
  systemctl restart apache2
segura_velasco@seguravelasco:~$ sudo systemctl restart apache2
segura_velasco@seguravelasco:~$
```



The screenshot shows a terminal window titled "Linux Apache [Corriendo] - Oracle VM VirtualBox". The terminal is running the nano text editor on the file `/etc/apache2/mods-enabled/info.conf`. The content of the file is as follows:

```
<IfModule mod_info.c>

    # Allow remote server configuration reports, with the URL of
    # http://servername/server-info (requires that mod_info.c be loaded).
    # Uncomment and change the "192.0.2.0/24" to allow access from other hosts.
    #
    <Location /server-info>
        SetHandler server-info
        Require local
        Require ip 172.26.0.46
    </Location>
</IfModule>

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

The terminal window has a menu bar with "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". The status bar at the bottom shows the time as 14:01 and the date as 07/12/2022.

Server Information

No es seguro | 172.26.245.1/server-info

Google Moodle Centros IES Velazquez

## Apache Server Information

Subpages:  
[Configuration Files](#), [Server Settings](#), [Module List](#), [Active Hooks](#), [Available Providers](#)

Sections:  
[Loaded Modules](#), [Server Settings](#), [Startup Hooks](#), [Request Hooks](#), [Other Hooks](#), [Providers](#)

### Loaded Modules

[core.c](#), [event.c](#), [http\\_core.c](#), [mod\\_access\\_compat.c](#), [mod\\_alias.c](#), [mod\\_auth\\_basic.c](#), [mod\\_auth\\_digest.c](#), [mod\\_authn\\_core.c](#), [mod\\_authn\\_file.c](#), [mod\\_authz\\_core.c](#), [mod\\_authz\\_host.c](#), [mod\\_authz\\_user.c](#), [mod\\_autoindex.c](#), [mod\\_deflate.c](#), [mod\\_dir.c](#), [mod\\_env.c](#), [mod\\_filter.c](#), [mod\\_info.c](#), [mod\\_log\\_config.c](#), [mod\\_logio.c](#), [mod\\_mime.c](#), [mod\\_negotiation.c](#), [mod\\_reqtimeout.c](#), [mod\\_setenvif.c](#), [mod\\_so.c](#), [mod\\_status.c](#), [mod\\_unixd.c](#), [mod\\_userdir.c](#), [mod\\_version.c](#), [mod\\_watchdog.c](#),

### Server Settings

Server Version: Apache/2.4.41 (Ubuntu)  
Server Built: 2022-06-14T13:30:55  
Server loaded APR Version: 1.6.5  
Compiled with APR Version: 1.6.5  
Server loaded APU Version: 1.6.1  
Compiled with APU Version: 1.6.1  
Module Magic Number: 20120211:88  
Hostname/port: 172.26.245.1:80  
Timeouts: connection: 300 keep-alive: 5  
MPM Name: event  
MPM Information: Max Daemons: 1 Threaded: yes Forked: yes  
Server Architecture: 64-bit  
Server Root: /etc/apache2  
Config File: /etc/apache2/apache2.conf  
Server Built With:  
-D APR\_HAS\_SENDFILE

14:02  
07/12/2022

## G) Webalizer

Otra forma de monitorizar nuestro servidor apache es mediante aplicaciones analizadoras de logs, como es el caso de `Webalizer`. Esta aplicación se puede instalar en nuestro servidor y a partir de los archivos logs te crea unas estadísticas que puedes consultar en formato html.

**PASO 1)** En tu servidor Linux, instala la aplicación Webalizer (usa `apt-get install`, pero antes actualiza el servidor Linux).

**PASO 2)** Una vez instalado se habrá creado un directorio para la aplicación en el **directorio `/etc/`**. Abre el fichero de configuración de `webalizer`, ¿de qué fichero log coge los datos para hacer las estadísticas? ¿es correcta la ruta y el nombre del fichero? Si no es así, modifícala.

El "LogFile" por defecto es `"/var/log/apache2/access.log.1"`... pero no es correcta, y eso la arreglamos quitando el "1".

**PASO 3)** La instalación también implica la creación del recurso que se servirá desde el navegador, ¿Dónde está este fichero? ¿Es correcta la ubicación para servirlo? **Si no es así, muévelo a la ubicación correcta.**

Por defecto tenemos `"/var/www/webalizer"`, y la debemos cambiar a `"/var/www/html/webalizer"`.

Podemos notar que una vez se descargó Webalizer **la ruta por defecto donde queda almacenado es `/var/www/webalizer`** y este parámetro **debemos moverlo a la ruta `/var/www/html`** para que la sincronización entre Apache y Webalizer sea correcta. Para realizar este proceso simplemente ejecutamos lo siguiente:

```
sudo mv /var/www/webalizer /var/www/html/
```

A continuación, vamos a **editar el archivo de configuración de Webalizer** introduce la siguiente instrucción:

```
sudo nano /etc/webalizer/webalizer.conf
```

```
GNU nano 4.8 /etc/webalizer/webalizer.conf
# Sample Webalizer configuration file
# Copyright 1997-2013 by Bradford L. Barrett
#
# Distributed under the GNU General Public License. See the
# files "Copyright" and "COPYING" provided with the webalizer
# distribution for additional information.
#
# This is a sample configuration file for the Webalizer (ver 2.23)
# Lines starting with pound signs '#' are comment lines and are
# ignored. Blank lines are skipped as well. Other lines are considered
# as configuration lines, and have the form "ConfigOption Value" where
# ConfigOption is a valid configuration keyword, and Value is the value
# to assign that configuration option. Invalid keyword/values are
# ignored, with appropriate warnings being displayed. There must be
# at least one space or tab between the keyword and its value.
#
# As of version 0.98, The Webalizer will look for a 'default' configuration
# file named "webalizer.conf" in the current directory, and if not found
# there, will look for "/etc/webalizer.conf".

# LogFile defines the web server log file to use. If not specified
# here or on the command line, input will default to STDIN. If
# the log filename ends in '.gz' (a gzip compressed file), or '.bz2'
# (bzip2 compressed file), it will be decompressed on the fly as it
# is being read.

LogFile /var/log/apache2/access.log.1

# LogType defines the log type being processed. Normally, the Webalizer
# expects a CLF or Combined web server log as input. Using this option,
# you can process ftp logs (xferlog as produced by wu-ftp and others),
# Squid native logs or W3C extended format web logs. Values can be 'clf',
# 'ftp', 'squid' or 'w3c'. The default is 'clf'.

#LogType          clf

# OutputDir is where you want to put the output files. This should
# should be a full path name, however relative ones might work as well.
# If no output directory is specified, the current directory will be used.

OutputDir /var/www/html/webalizer

# HistoryName allows you to specify the name of the history file produced
# by the Webalizer. The history file keeps the data for previous months,
# and is used for generating the main HTML page (index.html). The default
# is a file named "webalizer.hist", stored in the output directory being
# used. The name can include a path, which will be relative to the output
# directory unless absolute (starts with a leading '/').

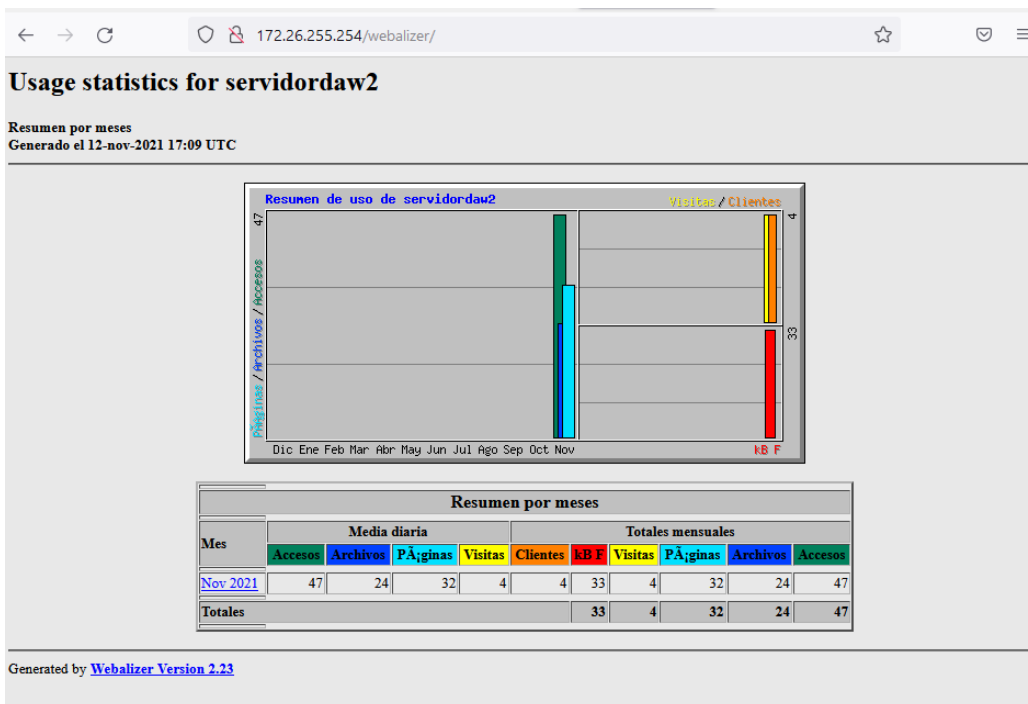
#HistoryName      webalizer.hist
```

**PASO 4)** Lanza el programa (con permisos de administrador) para que lea el fichero de log correspondiente y genere el documento `html` con las estadísticas.

**sudo webalizer**

**PASO 5)** Accede al recurso `/webalizer/` desde tu máquina física.





Toma una captura de los pasos 2 y 5.

```
environment magic rc4.d vtrgb
ethertypes magic.mime rc5.d webalizer
fonts mailcap rc6.d wgetrc
fstab mailcap.order rc8.d X11
fuse.conf manpath.config resolv.conf xattr.conf
fwupd mdadm rmt xdg
gal.conf mime.types rpc zsh_command_not_found
```

```
LogFile /var/log/apache2/access.log.1

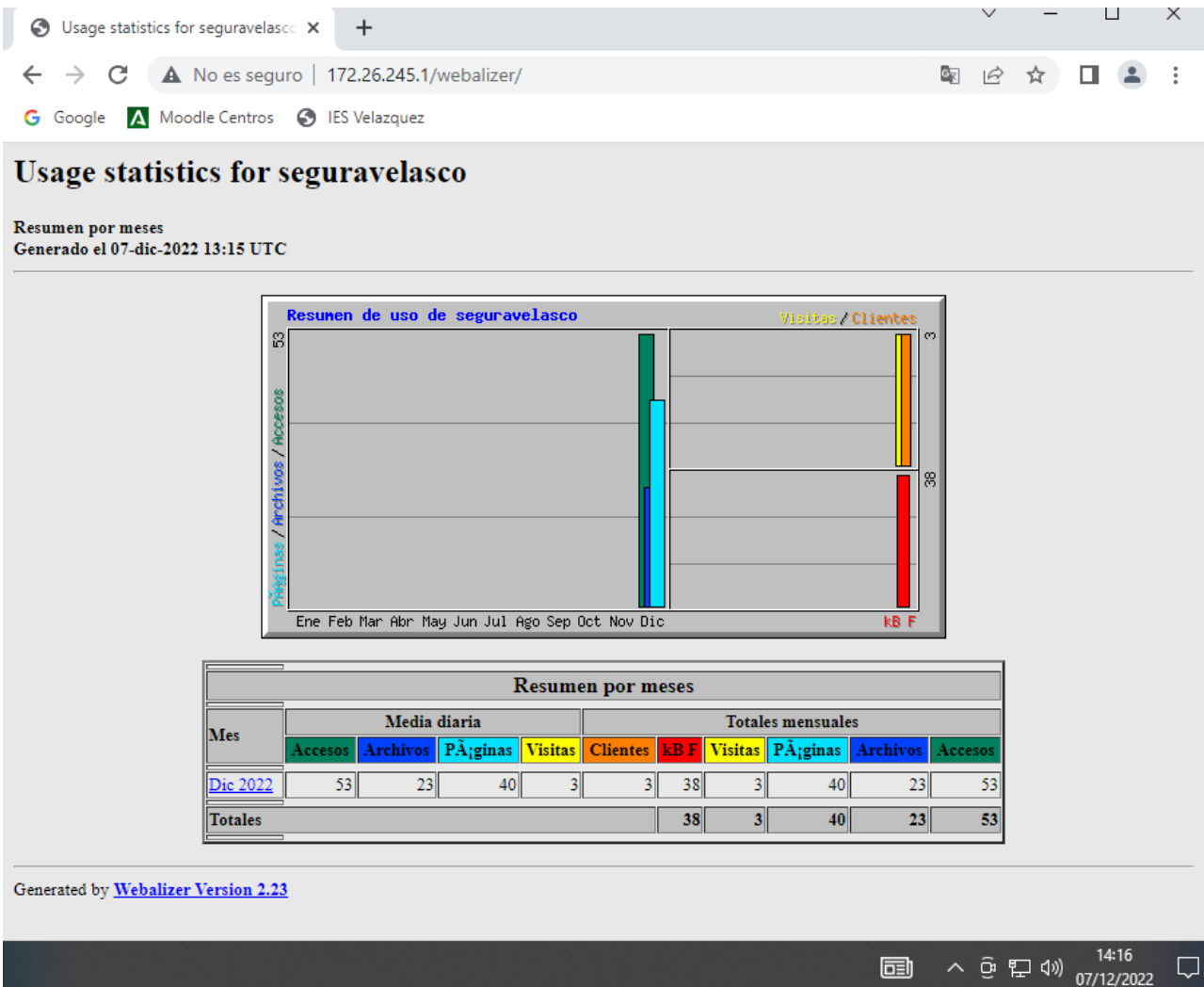
# LogType defines the log type being processed. Normally, the Webalizer
# expects a CLF or Combined web server log as input. Using this option,
# you can process ftp logs (xferlog as produced by wu-ftp and others),

[ Read 809 lines ]
Get Help Write Out Where Is Cut Text Justify Cur Pos M-U Undo
Exit Read File Replace Paste Text To Spell Go To Line M-E Redo
```

```
#LogType clf

# OutputDir is where you want to put the output files. This should
# should be a full path name, however relative ones might work as well.
# If no output directory is specified, the current directory will be used.

OutputDir /var/www/html/webalizer
```



### H) GitHub

Sube el documento al repositorio llamado Despliegue a la carpeta correspondiente.

Toma capturas de pantalla de los comandos utilizados y del repositorio de la página Web.

Link al repositorio Github: <https://github.com/Dansegvel/PracticaApacheLinux.git>