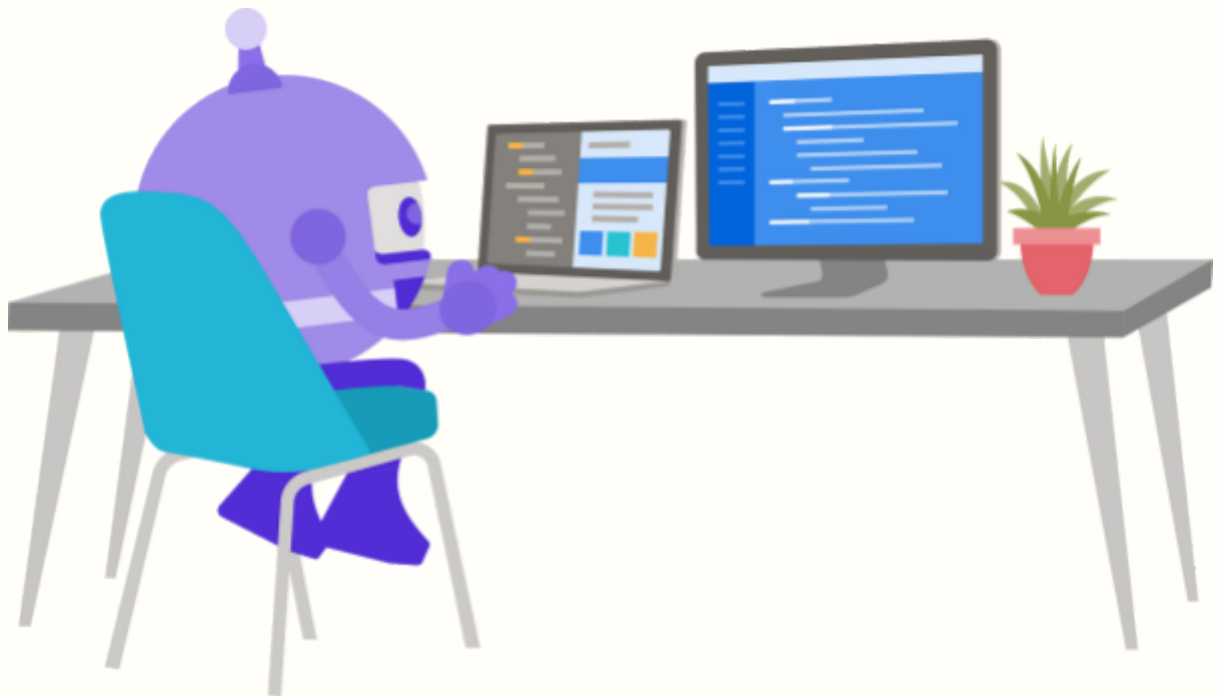


Administración de Apache II

Módulos



A) Módulos en Linux.....	3
A.1) Módulos.....	4
A.2) Módulo userdir.....	8
A.3) Módulo userdir en el servidor de clase.....	11
B) Control de acceso por IP y nombre de dominio.....	13
C) Autenticación y autorización Basic y Digest.....	17
C.1) Autenticación Basic.....	18
C.2) Autenticación Digest.....	23
D) Ficheros .htaccess.....	27
E) Ficheros de registros (logs).....	31
F) Módulos status e info.....	34
G) Webalizer.....	39
H) GitHub.....	42

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.

Para ver cuales son los módulos estáticos usaremos:

```
sudo apache2ctl -l
```

```
servidordgg@servidordgg:~$ sudo apache2ctl -l
[sudo] password for servidordgg:
Compiled in modules:
  core.c
  mod_so.c
  mod_watchdog.c
  http_core.c
  mod_log_config.c
  mod_logio.c
  mod_version.c
  mod_unixd.c
servidordgg@servidordgg:~$
```

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

Para comprobar los módulos que se han cargado dinámicamente usaremos:

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

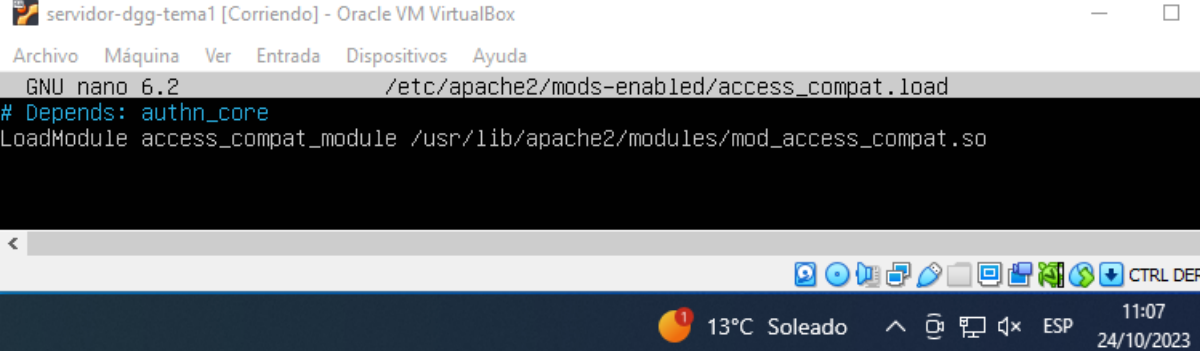
```
servidor-dgg-tema1 [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
servidordgg@servidordgg:/$ ls /etc/apache2/mods-enabled/
access_compat.load  authz_core.load  deflate.load  mime.load  reqtimeout.load
alias.conf          authz_host.load  dir.conf     mpm_event.conf  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
servidordgg@servidordgg:/$
```

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?

Para editar un archivo usaremos el siguiente comando:

```
sudo nano /etc/apache2/mods-enabled/access_compat.load
```

La extensión que usan los archivos donde está el código es **.so**



```
servidor-dgg-tema1 [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 6.2 /etc/apache2/mods-enabled/access_compat.load
# Depends: authn_core
LoadModule access_compat_module /usr/lib/apache2/modules/mod_access_compat.so
```

The screenshot shows a terminal window titled 'servidor-dgg-tema1 [Corriendo] - Oracle VM VirtualBox'. Inside, the nano text editor is open, editing the file '/etc/apache2/mods-enabled/access_compat.load'. The editor's status bar at the top indicates 'GNU nano 6.2'. The file content shows a comment '# Depends: authn_core' and a 'LoadModule' directive: 'LoadModule access_compat_module /usr/lib/apache2/modules/mod_access_compat.so'. The terminal window has a standard Linux desktop environment at the bottom with system icons, a taskbar showing '13°C Soleado', and a clock indicating '11:07' on '24/10/2023'.

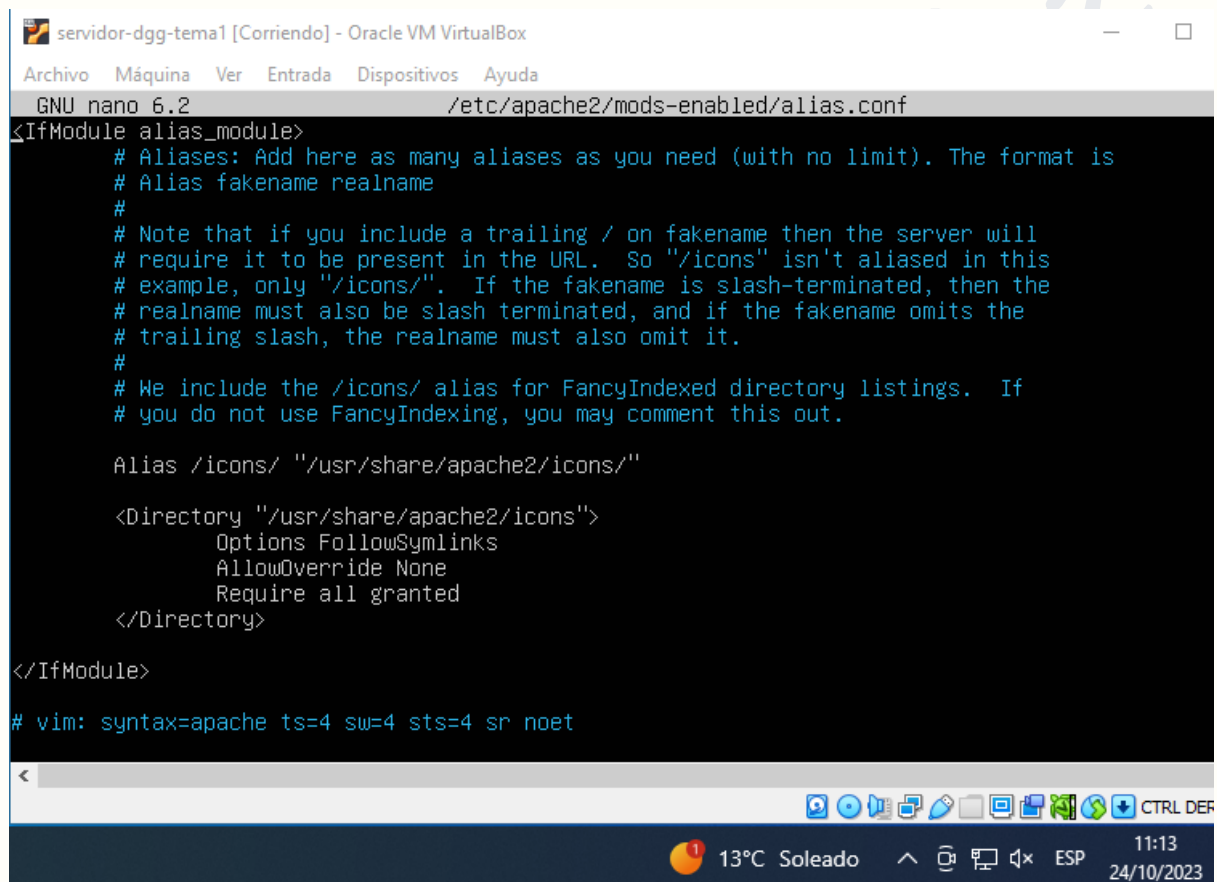
PASO 4) Edita uno de los archivos .conf y observa cómo se añaden directivas dentro del módulo.

Para ello usaremos el siguiente comando:

```
sudo nano /etc/apache2/mods-enabled/alias.conf
```

¿Qué etiquetas se utilizan en estos archivos?

Como podemos observar, usan etiquetas **XML**



```
servidor-dgg-tema1 [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 6.2 /etc/apache2/mods-enabled/alias.conf
<IfModule alias_module>
# Aliases: Add here as many aliases as you need (with no limit). The format is
# Alias fakename realname
#
# Note that if you include a trailing / on fakename then the server will
# require it to be present in the URL.  So "/icons" isn't aliased in this
# example, only "/icons/".  If the fakename is slash-terminated, then the
# realname must also be slash terminated, and if the fakename omits the
# trailing slash, the realname must also omit it.
#
# We include the /icons/ alias for FancyIndexed directory listings.  If
# you do not use FancyIndexing, you may comment this out.

Alias /icons/ "/usr/share/apache2/icons/"

<Directory "/usr/share/apache2/icons">
    Options FollowSymlinks
    AllowOverride None
    Require all granted
</Directory>
</IfModule>

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

PASO 5) Consulta el directorio `/usr/lib/apache2/modules/` ¿qué archivos contiene?

Para ello usaremos el comando:

```
ls -l /usr/lib/apache2/modules/
```

Este directorio contiene archivos **.so**

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

David González

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?

Para comprobar si el módulo está habilitado usaremos el siguiente comando:

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

No está habilitado

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

Para habilitarlo deberemos usar el siguiente comando:

```
sudo a2enmod userdir
```

PASO 3) Verifica ahora si el módulo está habilitado.

Para ello usaremos el mismo comando del paso 1

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

```
servidordgg@servidordgg:~$ ls /etc/apache2/mods-enabled/
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
servidordgg@servidordgg:~$ _
```



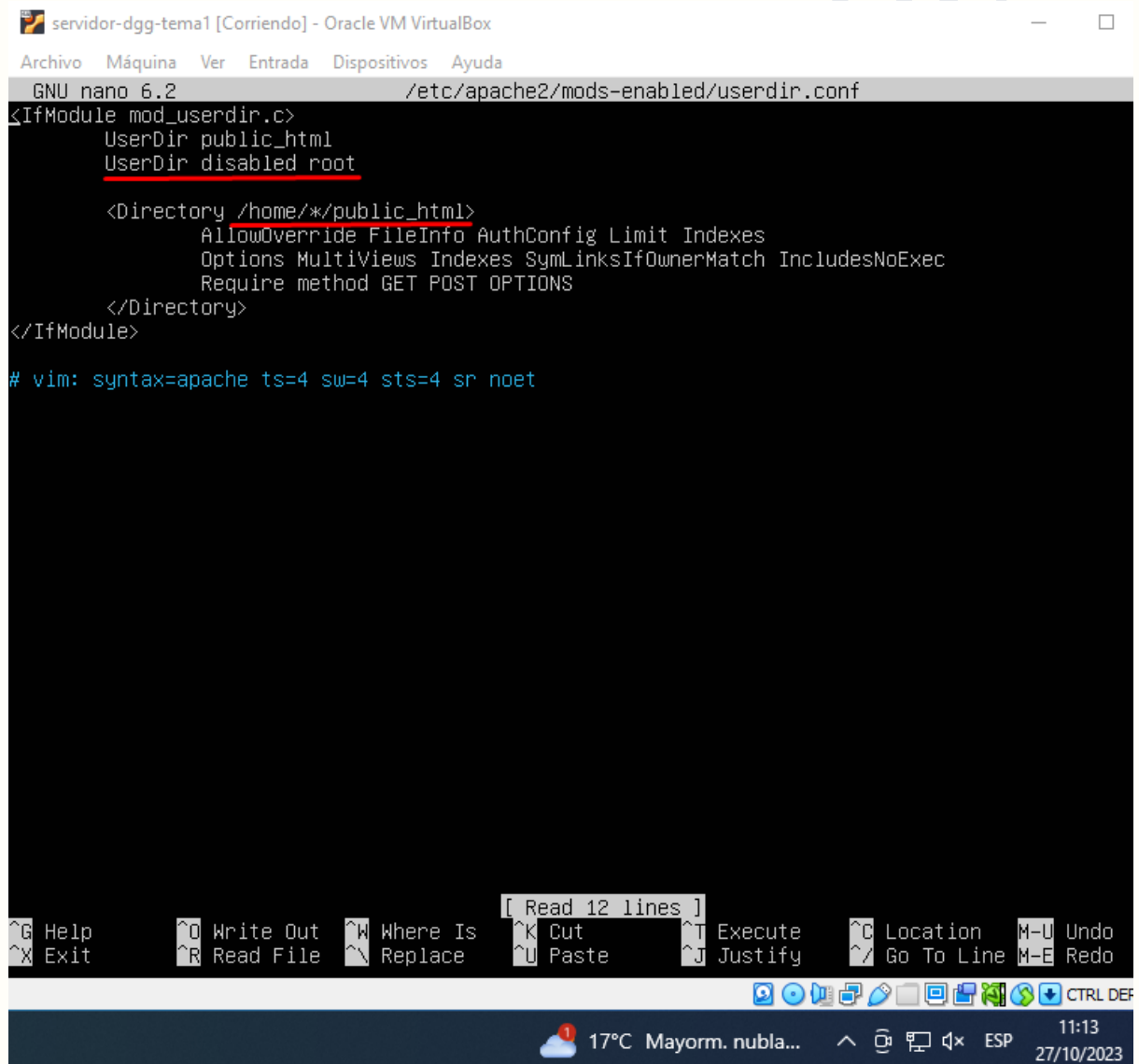
PASO 4) Reinicia el servidor para que los cambios tengan efecto.

Reiniciado 😊

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?

Para ello usaremos el siguiente comando:

```
sudo nano /etc/apache2/mods-enabled/userdir.conf
```



```
servidor-dgg-tema1 [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 6.2 /etc/apache2/mods-enabled/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

[ Read 12 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
CTRL DEF
17°C Mayorm. nubla... 11:13
27/10/2023
```

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.

Creados 😊

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

The screenshot shows a web browser window with two tabs. The active tab displays the directory index for /~profe. The page title is "Index of /~profe". Below the title is a table with columns: Name, Last modified, Size, and Description. The table lists two items: "Parent Directory" and "personal.html". The "personal.html" file was last modified on 2016-11-22 19:55 and has a size of 38 bytes. Below the table, it says "Apache/2.4.18 (Ubuntu) Server at 192.168.1.151 Port 80".

The second tab shows the directory index for /~servidordgg. The page title is "Index of /~servidordgg". Below the title is a table with columns: Name, Last modified, Size, and Description. The table lists two items: "Parent Directory" and "personal.html". The "personal.html" file was last modified on 2023-10-27 09:22 and has a size of 48 bytes. Below the table, it says "Apache/2.4.52 (Ubuntu) Server at 192.168.1.205 Port 80".

The browser's address bar shows the URL "192.168.1.205/~servidordgg/". The browser's status bar at the bottom shows the temperature as 18°C, the weather as Soleado, and the time as 12:13 on 27/10/2023.

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)

A.3) Módulo userdir en el servidor de clase

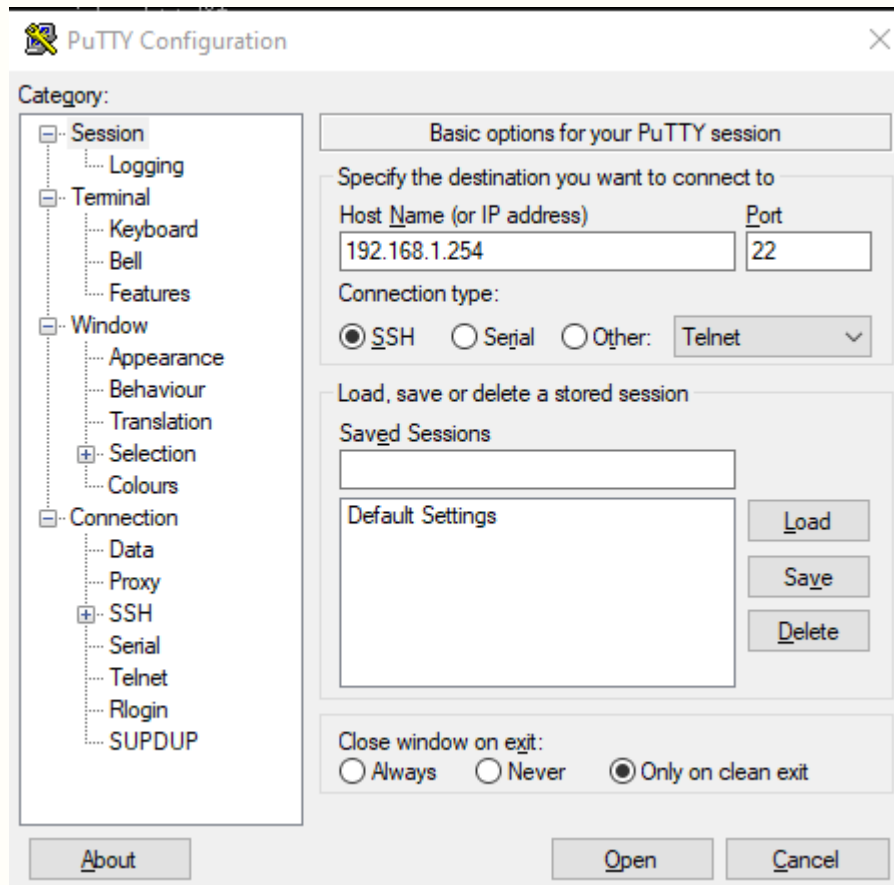
En el servidor del aula todos tenéis un usuario y una contraseña para entrar.

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

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

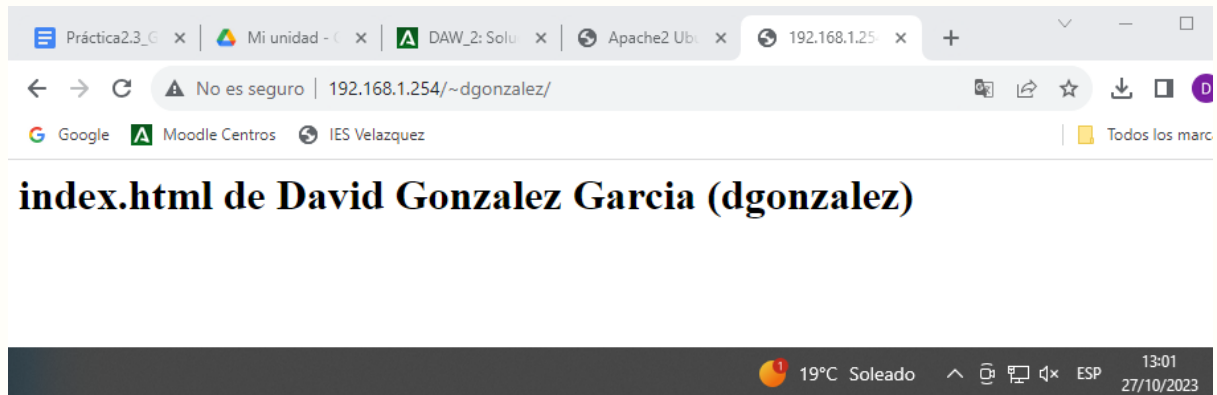
PASO 1) Accede al servidor a través de Putty. IP: 192.168.1.254

Una vez he instalada la aplicación PuTTY, he insertado la ip correspondiente:



PASO 2) Da los pasos necesarios para que al acceder a <http://192.168.1.254/~agutierrez> se vea tu página web en el servidor.

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



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

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?

Para comprobar si está activado usaremos:

```
$ ls /etc/apache2/mods-enabled/
```

Si, está habilitado.

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.

Para ello usaremos los siguientes comandos:

```
$ mkdir /var/www/html/david/  
$ nano /var/www/html/david/david.html
```

PASO 3) Edita el fichero de configuración `/etc/apache2/sites-available/000-default.conf` y añade la directiva `Directory` para el recurso creado anteriormente.

Para ello usaremos:

```
$ nano /etc/apache2/sites-available/000-default.conf
```

```
<Directory /var/www/html/david/>
</Directory>
</VirtualHost>

root@servidordgg:/home/servidordgg# _
```

PASO 4) 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).

Para ello volveremos a usar:

```
$ nano /etc/apache2/sites-available/000-default.conf
```

```
<Directory /var/www/html/david/>
    Require ip 192.168.1.176
    Require host AULA43-PC06
</Directory>

root@servidordgg:/home/servidordgg#
```

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

Para ello usaremos:

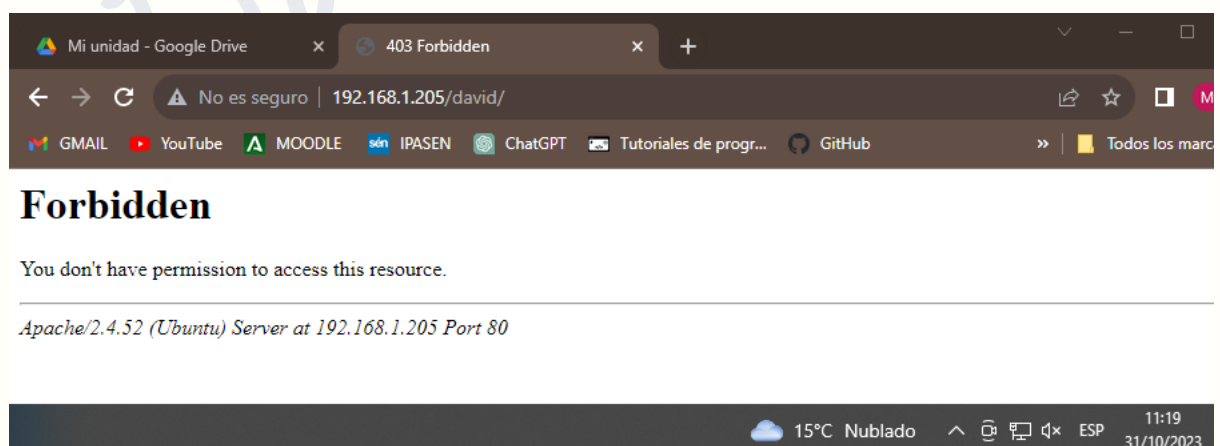
```
$ systemctl restart apache2
```

```
root@servidordgg:/home/servidordgg# systemctl restart apache2
root@servidordgg:/home/servidordgg# _
```

PASO 6) Abre un navegador desde tu máquina física e intenta acceder al recurso /tuNombre/ y comprueba que se puede.



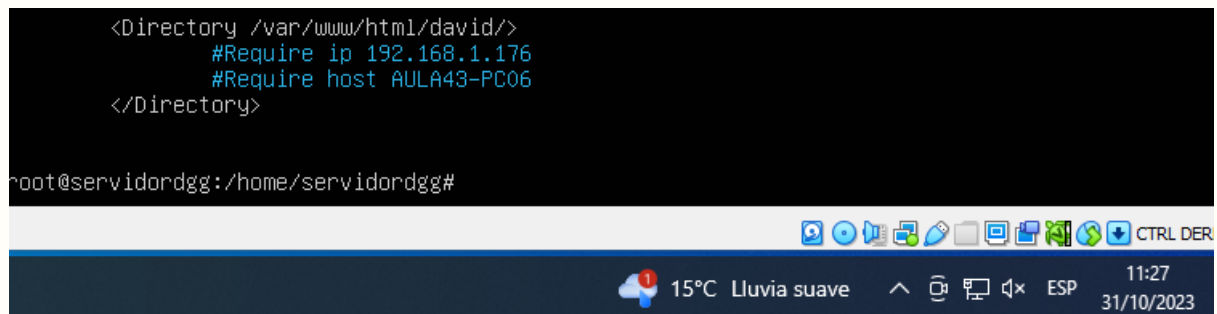
PASO 7) Abre un navegador desde la máquina del compañero e intenta acceder al recurso /tuNombre/ y comprueba que no se puede.



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.

```
<Directory /var/www/html/david/>
    #Require ip 192.168.1.176
    #Require host AULA43-PC06
</Directory>

root@servidordgg:/home/servidordgg#
```



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

Para ello usamos:

```
$ systemctl restart apache2
```

PASO 9) Abre un navegador desde la máquina del compañero e intenta acceder al recurso /tuNombre/ y comprueba que ahora sí se puede.

No funciona

Toma una captura de los pasos 3,4,5, 6, 7 y 9.

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 a 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:
 - o **AuthType**: tipo de autorización
 - o **AuthName**: nombre de la autorización cuando el cliente reciba el mensaje
 - o **AuthUserFile**: localización del fichero donde están los usuarios que pueden autenticarse
 - o **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:
 - o **AuthType**: tipo de autorización
 - o **AuthName**: nombre de la autorización cuando el cliente reciba el mensaje
 - o **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
 - o **AuthUserFile**: localización del fichero donde están los usuarios que pueden autenticarse
 - o **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.

Para ello usaremos el siguiente comando:

```
$ sudo ls /etc/apache2/mods-enabled/
```

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.

Para ello usaremos los siguientes comandos:

Ya hemos hecho este paso anteriormente, [AQUÍ](#)

```
$ sudo mkdir /var/www/html/david
$ sudo nano /var/www/html/david/david.html
```

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 primera vez que se invoca el comando se
# utiliza a 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
```

```
servidordgg@servidordgg:~$ sudo htpasswd -c /etc/apache2/passwd gonzalez
[sudo] password for servidordgg:
New password:
Re-type new password:
Adding password for user gonzalez
servidordgg@servidordgg:~$ sudo htpasswd /etc/apache2/passwd garcia
New password:
Re-type new password:
Adding password for user garcia
servidordgg@servidordgg:~$
```

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).

Para ello usaremos el comando:

```
$ sudo nano /etc/apache2/sites-available/000-default.conf
```

```
# 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

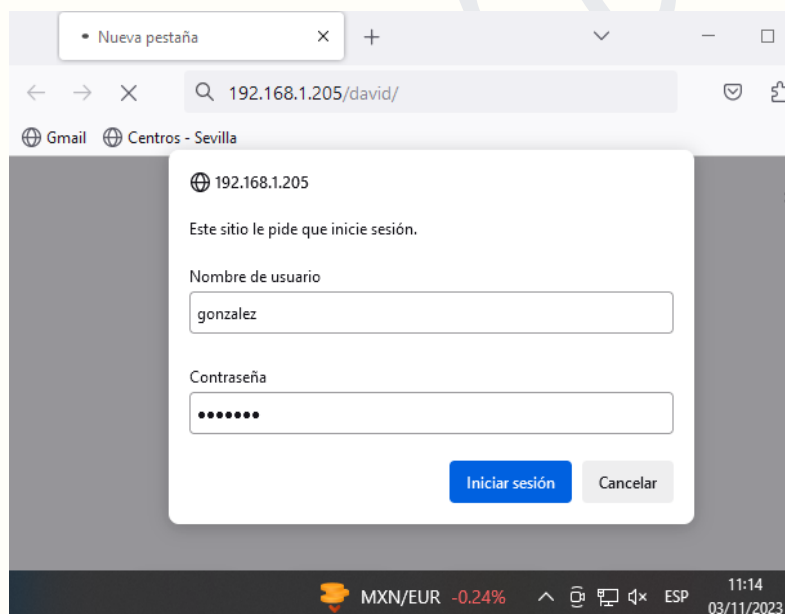
<Directory /var/www/html/david/>
    #Require ip 192.168.1.176
    #Require host AULA43-PC06
    AuthType Basic
    AuthName "Acceso restringido"
    AuthUserFile /etc/apache2/passwd
    Require user gonzalez garcia_
</Directory>
```

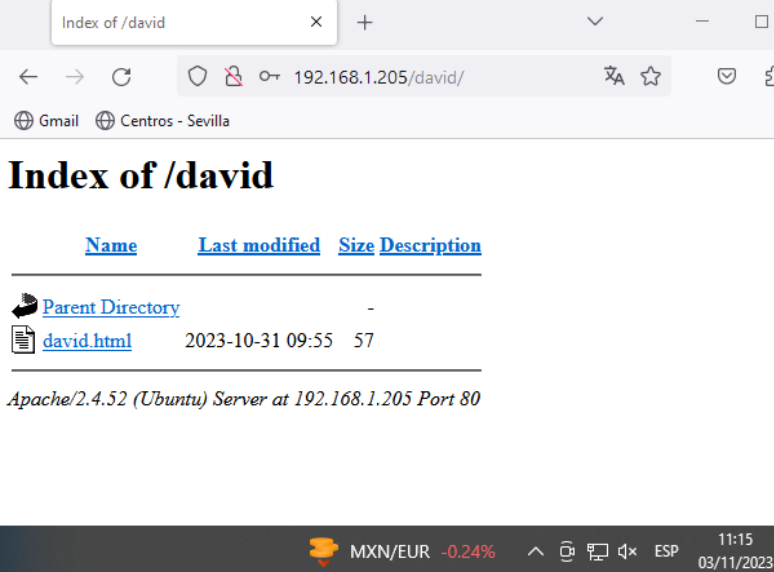
PASO 5) Reinicia el servidor para que los cambios tengan efecto.

Para ello usaremos:

```
$ systemctl restart apache2
```

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





The screenshot shows a web browser window with the address bar displaying "192.168.1.205/david/". The page title is "Index of /david". Below the title, there is a table with columns: Name, Last modified, Size, and Description. The table contains two entries: "Parent Directory" with a size of "-" and "david.html" with a last modified date of "2023-10-31 09:55" and a size of "57". Below the table, it says "Apache/2.4.52 (Ubuntu) Server at 192.168.1.205 Port 80". At the bottom of the browser window, there is a status bar showing "MXN/EUR -0.24%", "ESP", and the time "11:15 03/11/2023".

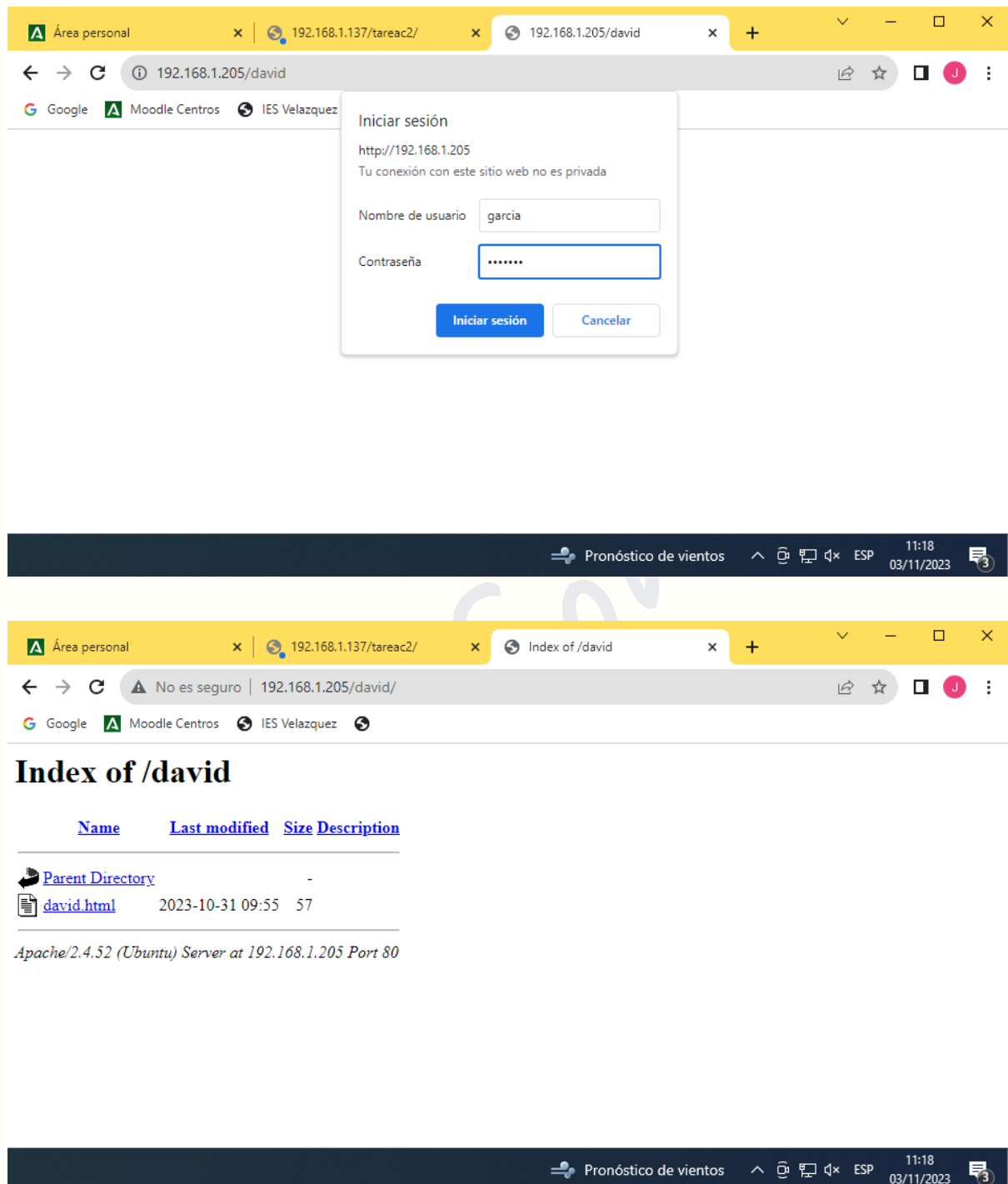
Index of /david

Name	Last modified	Size	Description
Parent Directory	-		
david.html	2023-10-31 09:55	57	

Apache/2.4.52 (Ubuntu) Server at 192.168.1.205 Port 80

MXN/EUR -0.24% ESP 11:15 03/11/2023

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 estos últimos una captura cuando sale el cuadro para autenticarte y luego una vez dentro del recurso /amigo).

C.2) Autenticación Digest

PASO 1) Comprueba si el módulo `auth_digest` está habilitado, si no lo está, habilítalo.

Para ello usaremos:

```
$ sudo ls /etc/apache2/mods-enabled/
```

Para activarlo usaremos:

```
$ a2enmod auth_digest
```

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.

Para ello usaremos:

```
$ sudo mkdir /var/www/html/tareac2  
$ sudo nano /var/www/html/david/tareac2.html
```

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.

Ejemplo: Amapola Gutierrez de la Vega:

gamapola
vamapola

```
# 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
```

```
servidordgg@servidordgg:~$ sudo htdigest -c /etc/apache2/digest realm gondavid
[sudo] password for servidordgg:
Adding password for gondavid in realm realm.
New password:
Re-type new password:
servidordgg@servidordgg:~$ sudo htdigest /etc/apache2/digest realm gardavid
Adding user gardavid in realm realm
New password:
Re-type new password:
servidordgg@servidordgg:~$
```

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”.

```
<Directory /var/www/html/tareac2/>
    AuthType Digest
    AuthName "realm"
    AuthDigestProvider file
    AuthUserFile /etc/apache2/digest
    Require user gondavid gardavid
</Directory>
/VirtualHost>
```

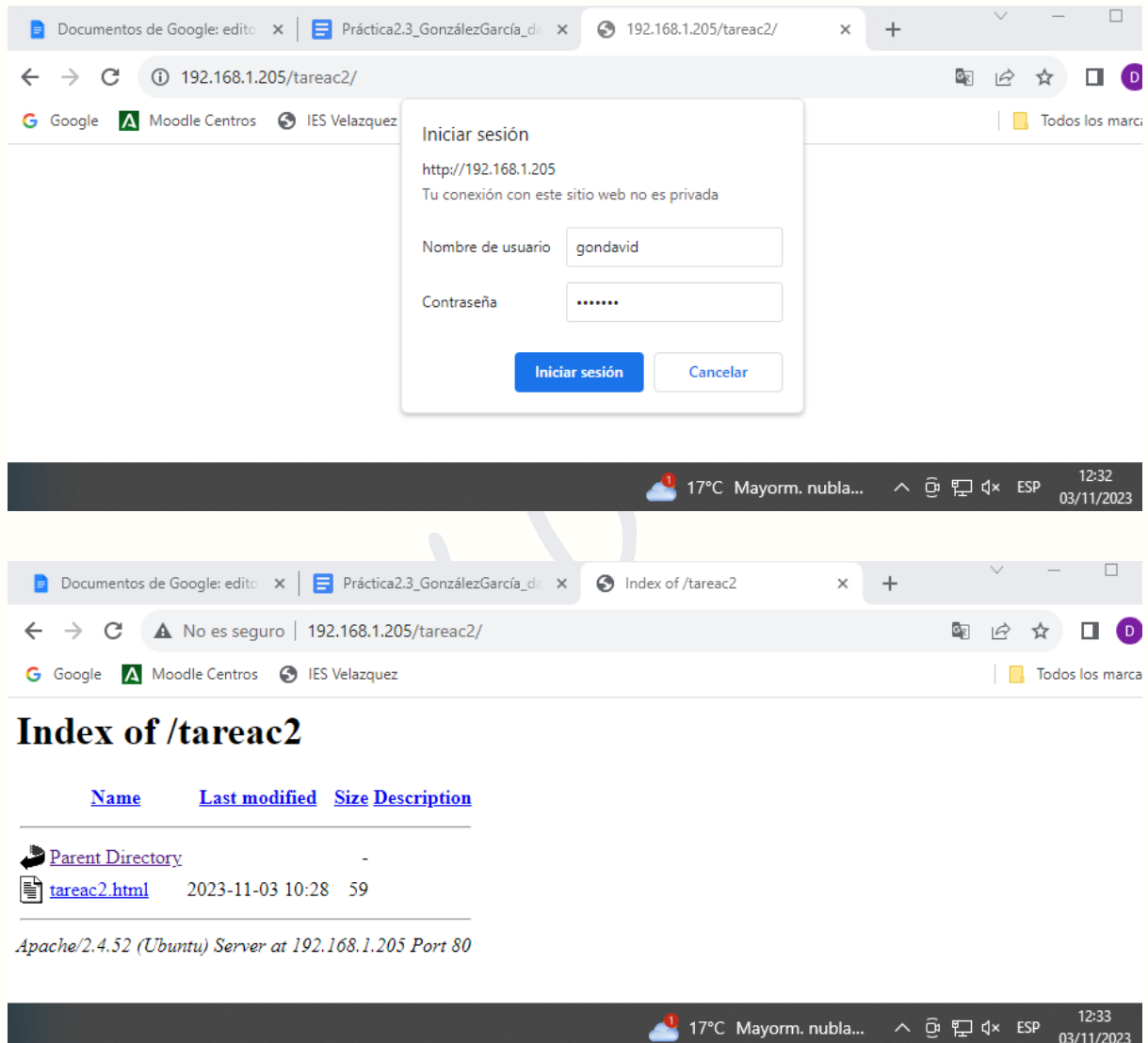
```
servidordgg@servidordgg:~$ _
```


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

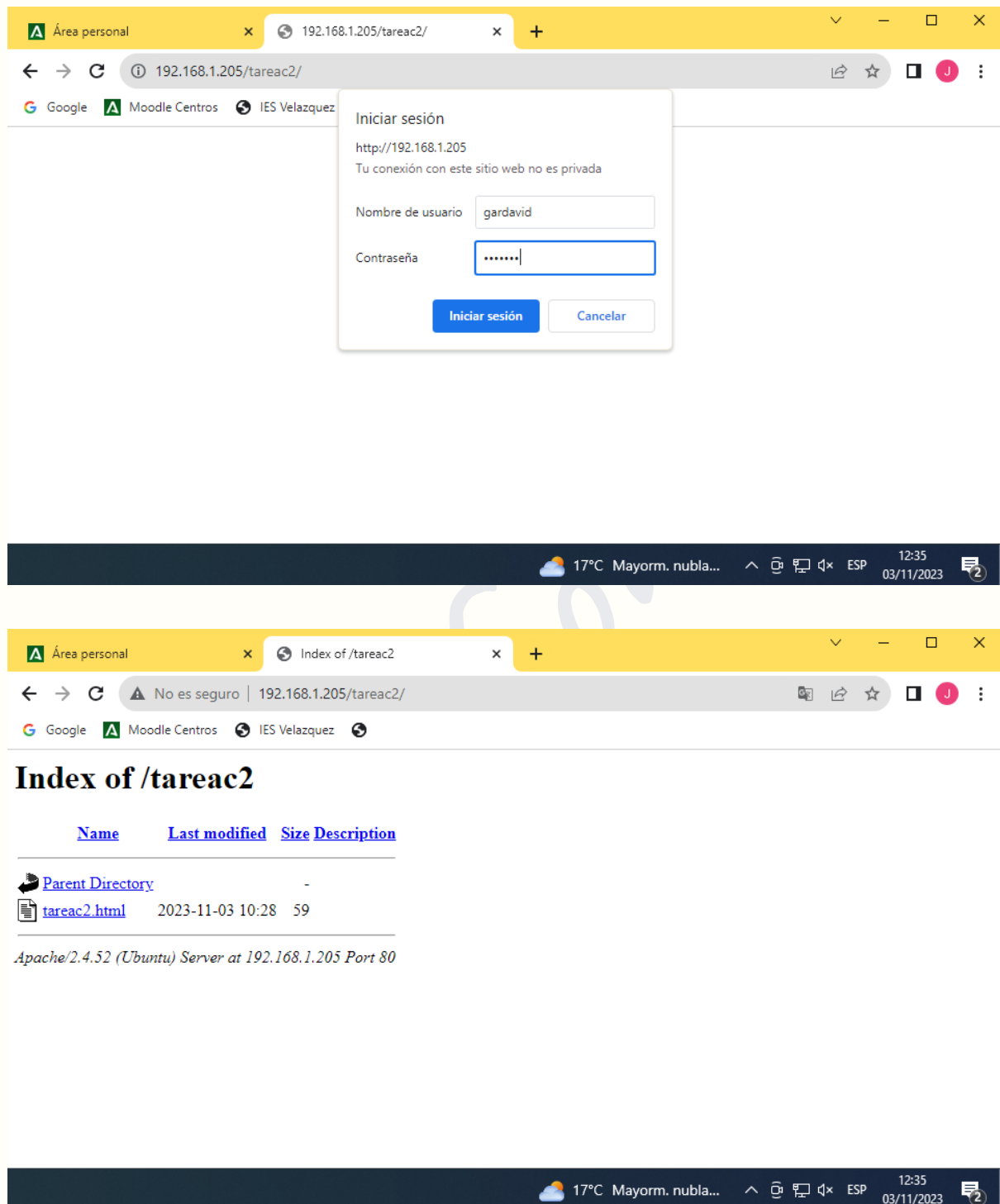
Para ello usaremos:

```
$ systemctl restart apache2
```

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 estos últimos una captura cuando sale el cuadro para autenticarte y luego una vez dentro del recurso /primo).

D) Ficheros .htaccess

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.

Para ello usaremos:

```
$ sudo adduser useraccess
```

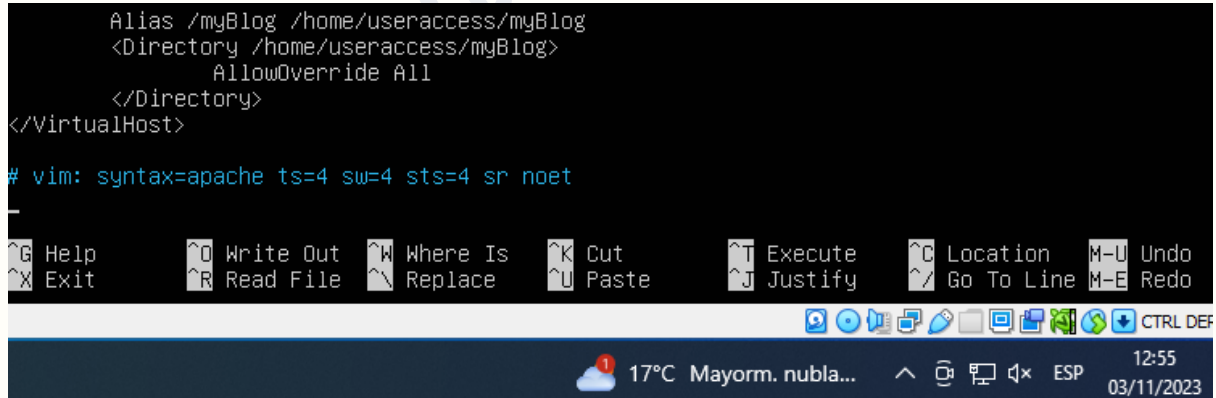
PASO 2) Abre el fichero de configuración 000-default y crea el alias myBlog dentro de la carpeta personal del nuevo usuario useraccess. Deja como única directiva AllowOverride All.

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

Para ello usaremos:

```
$ sudo nano /etc/apache2/sites-available/000-default.conf
```

```
Alias /myBlog /home/useraccess/myBlog
<Directory /home/useraccess/myBlog>
    AllowOverride All
</Directory>
</VirtualHost>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```



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

Para ello usaremos:

```
$ systemctl restart apache2
```

PASO 4) Inicia sesión con el nuevo usuario useraccess.

Para ello usaremos:

```
$ su 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.

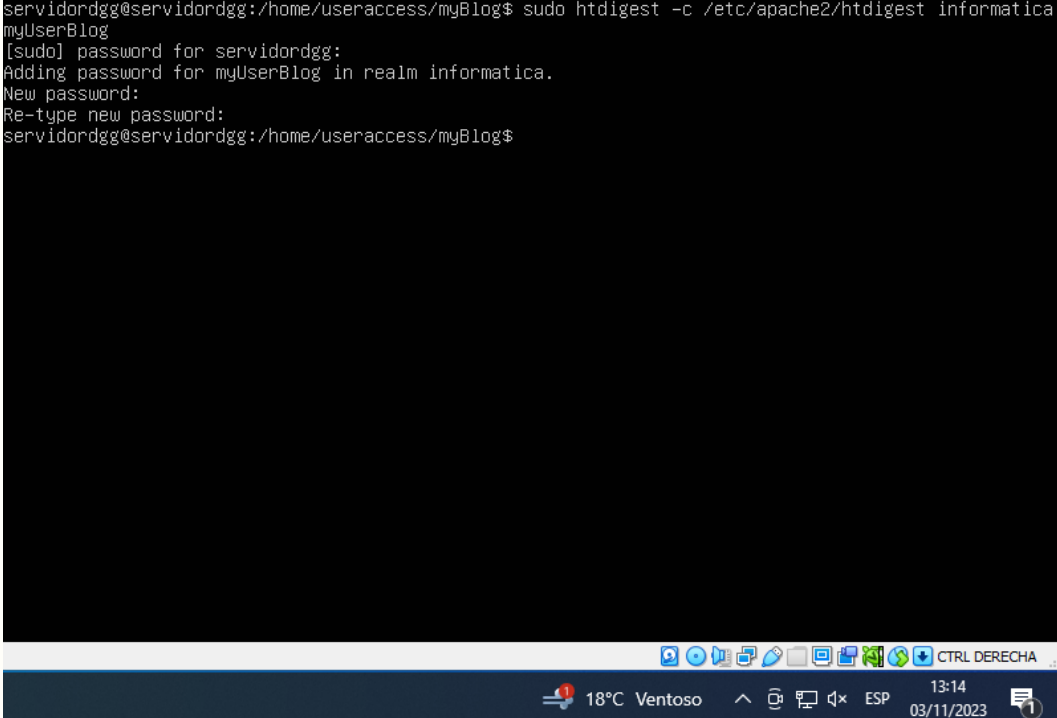
Para ello usaremos:

```
$ mkdir /home/useraccess/myBlog  
$ nano /home/useraccess/myBlog/myBlog.html
```

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).

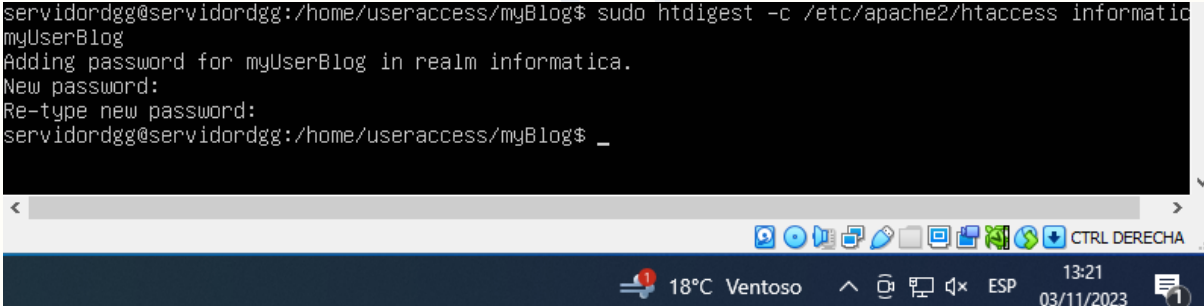
```
# La primera vez que se invoca el comando se  
# utiliza a opción -c para crear el fichero  
htdigest -c /etc/apache2/digest    informatica admin1  
  
# Añade un nuevo usuario al fichero  
Htdigest /etc/apache2/digest    informatica admin2  
  
# Borrar un nuevo usuario al fichero  
htdigest -D /etc/apache2/digest    informatica admin1
```

```
servidordgg@servidordgg:/home/useraccess/myBlog$ sudo htdigest -c /etc/apache2/htdigest informatica
myUserBlog
[sudo] password for servidordgg:
Adding password for myUserBlog in realm informatica.
New password:
Re-type new password:
servidordgg@servidordgg:/home/useraccess/myBlog$
```



PASO 7) Ahora tendremos que crear el fichero `.htaccess` (también dentro de `myBlog`). Dentro añadiremos las directivas necesarias para que se acceda sólo 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`).

```
servidordgg@servidordgg:/home/useraccess/myBlog$ sudo htdigest -c /etc/apache2/htaccess informatica
myUserBlog
Adding password for myUserBlog in realm informatica.
New password:
Re-type new password:
servidordgg@servidordgg:/home/useraccess/myBlog$ _
```



```
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.

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?

```
/var/log/apache2/error.log
```

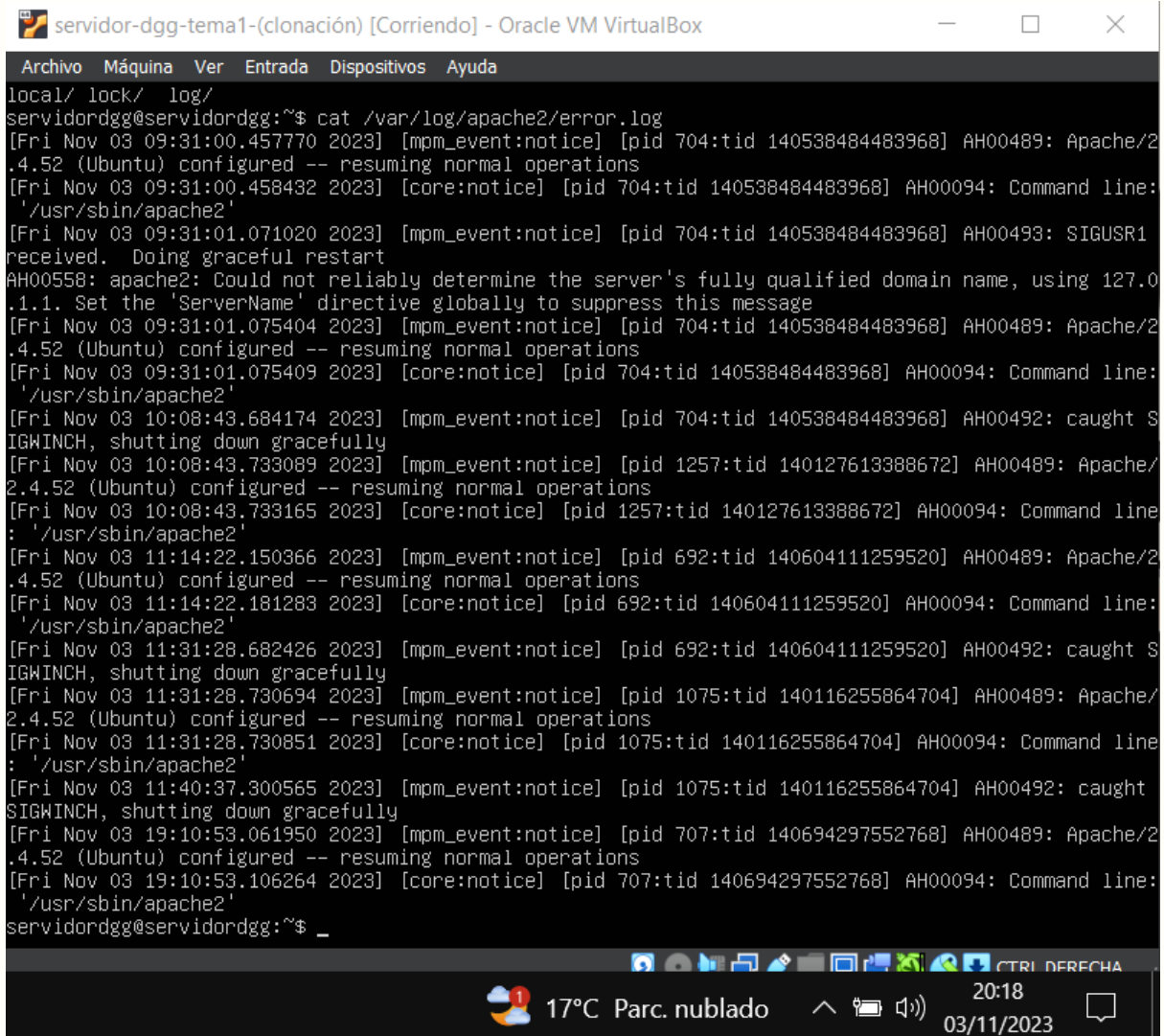
¿Qué directiva marca la ruta del archivo de los accesos? ¿Cuál es el fichero de logs de accesos?

```
/var/log/apache2/access.log
```

PASO 2) Consulta el log de errores

Para ello usaremos:

```
$ cat /var/log/apache2/error.log
```



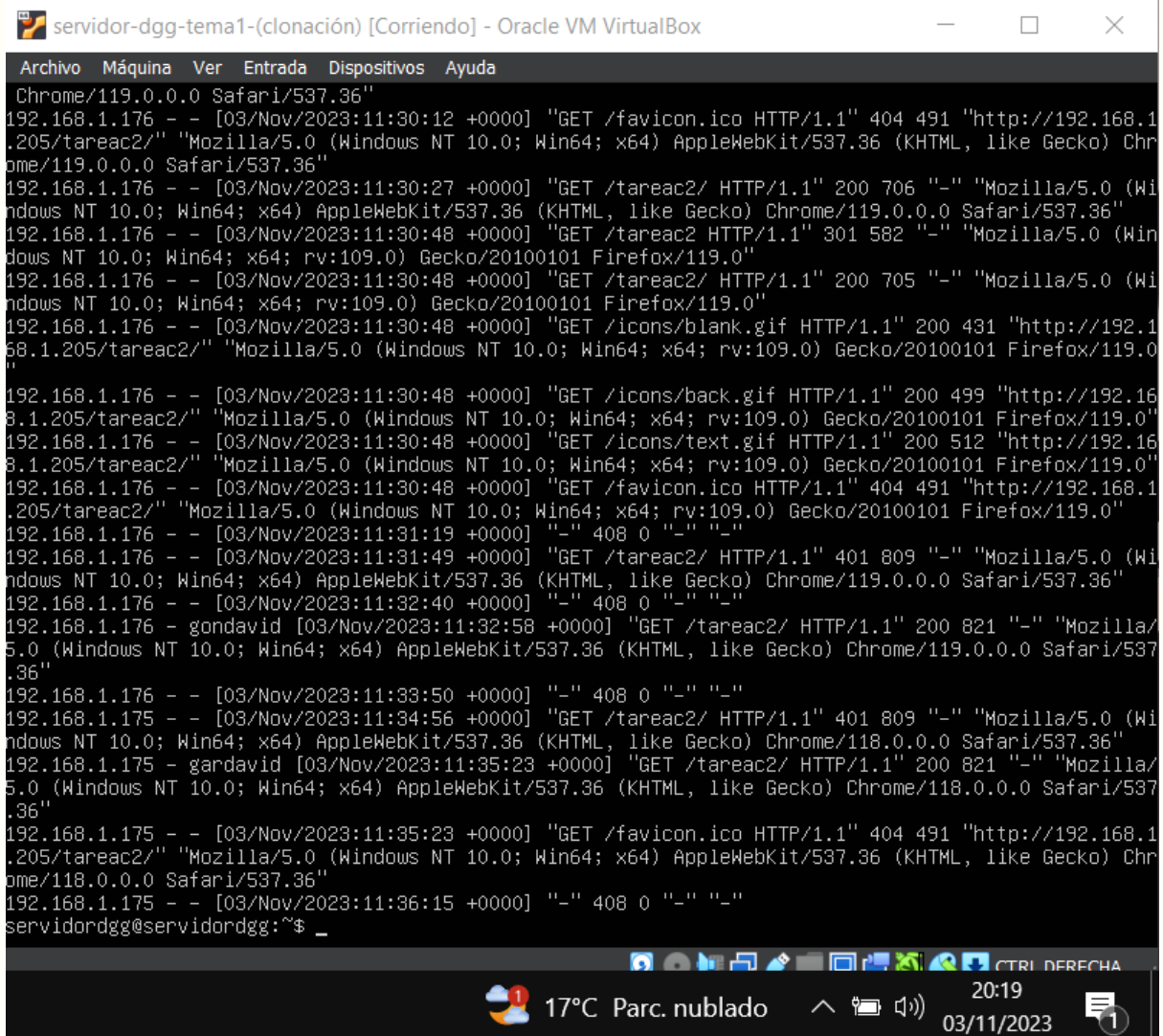
The screenshot shows a terminal window titled "servidor-dgg-tema1-(clonación) [Corriendo] - Oracle VM VirtualBox". The terminal displays the output of the command `cat /var/log/apache2/error.log`. The log contains several entries, including configuration messages, command line information, and graceful restarts. The entries are timestamped and include process IDs and thread IDs. The terminal also shows the system tray at the bottom with weather information (17°C, Parc. nublado) and the date (03/11/2023).

```
local/ lock/ log/
servidordgg@servidordgg:~$ cat /var/log/apache2/error.log
[Fri Nov 03 09:31:00.457770 2023] [mpm_event:notice] [pid 704:tid 140538484483968] AH00489: Apache/2
.4.52 (Ubuntu) configured -- resuming normal operations
[Fri Nov 03 09:31:00.458432 2023] [core:notice] [pid 704:tid 140538484483968] AH00094: Command line:
'/usr/sbin/apache2'
[Fri Nov 03 09:31:01.071020 2023] [mpm_event:notice] [pid 704:tid 140538484483968] AH00493: SIGUSR1
received. Doing graceful restart
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0
.1.1. Set the 'ServerName' directive globally to suppress this message
[Fri Nov 03 09:31:01.075404 2023] [mpm_event:notice] [pid 704:tid 140538484483968] AH00489: Apache/2
.4.52 (Ubuntu) configured -- resuming normal operations
[Fri Nov 03 09:31:01.075409 2023] [core:notice] [pid 704:tid 140538484483968] AH00094: Command line:
'/usr/sbin/apache2'
[Fri Nov 03 10:08:43.684174 2023] [mpm_event:notice] [pid 704:tid 140538484483968] AH00492: caught S
IGWINCH, shutting down gracefully
[Fri Nov 03 10:08:43.733089 2023] [mpm_event:notice] [pid 1257:tid 140127613388672] AH00489: Apache/
2.4.52 (Ubuntu) configured -- resuming normal operations
[Fri Nov 03 10:08:43.733165 2023] [core:notice] [pid 1257:tid 140127613388672] AH00094: Command line
: '/usr/sbin/apache2'
[Fri Nov 03 11:14:22.150366 2023] [mpm_event:notice] [pid 692:tid 140604111259520] AH00489: Apache/2
.4.52 (Ubuntu) configured -- resuming normal operations
[Fri Nov 03 11:14:22.181283 2023] [core:notice] [pid 692:tid 140604111259520] AH00094: Command line:
'/usr/sbin/apache2'
[Fri Nov 03 11:31:28.682426 2023] [mpm_event:notice] [pid 692:tid 140604111259520] AH00492: caught S
IGWINCH, shutting down gracefully
[Fri Nov 03 11:31:28.730694 2023] [mpm_event:notice] [pid 1075:tid 140116255864704] AH00489: Apache/
2.4.52 (Ubuntu) configured -- resuming normal operations
[Fri Nov 03 11:31:28.730851 2023] [core:notice] [pid 1075:tid 140116255864704] AH00094: Command line
: '/usr/sbin/apache2'
[Fri Nov 03 11:40:37.300565 2023] [mpm_event:notice] [pid 1075:tid 140116255864704] AH00492: caught
SIGWINCH, shutting down gracefully
[Fri Nov 03 19:10:53.061950 2023] [mpm_event:notice] [pid 707:tid 140694297552768] AH00489: Apache/2
.4.52 (Ubuntu) configured -- resuming normal operations
[Fri Nov 03 19:10:53.106264 2023] [core:notice] [pid 707:tid 140694297552768] AH00094: Command line:
'/usr/sbin/apache2'
servidordgg@servidordgg:~$ _
```


PASO 3) Consulta el log de accesos

Para ello usaremos:

```
$ cat /var/log/apache2/access.log
```



```
servidor-dgg-tema1-(clonación) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Chrome/119.0.0.0 Safari/537.36"
192.168.1.176 - - [03/Nov/2023:11:30:12 +0000] "GET /favicon.ico HTTP/1.1" 404 491 "http://192.168.1.205/tareac2/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36"
192.168.1.176 - - [03/Nov/2023:11:30:27 +0000] "GET /tareac2/ HTTP/1.1" 200 706 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36"
192.168.1.176 - - [03/Nov/2023:11:30:48 +0000] "GET /tareac2 HTTP/1.1" 301 582 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/119.0"
192.168.1.176 - - [03/Nov/2023:11:30:48 +0000] "GET /tareac2/ HTTP/1.1" 200 705 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/119.0"
192.168.1.176 - - [03/Nov/2023:11:30:48 +0000] "GET /icons/blank.gif HTTP/1.1" 200 431 "http://192.168.1.205/tareac2/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/119.0"
192.168.1.176 - - [03/Nov/2023:11:30:48 +0000] "GET /icons/back.gif HTTP/1.1" 200 499 "http://192.168.1.205/tareac2/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/119.0"
192.168.1.176 - - [03/Nov/2023:11:30:48 +0000] "GET /icons/text.gif HTTP/1.1" 200 512 "http://192.168.1.205/tareac2/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/119.0"
192.168.1.176 - - [03/Nov/2023:11:30:48 +0000] "GET /favicon.ico HTTP/1.1" 404 491 "http://192.168.1.205/tareac2/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/119.0"
192.168.1.176 - - [03/Nov/2023:11:31:19 +0000] "-" 408 0 "-" "-"
192.168.1.176 - - [03/Nov/2023:11:31:49 +0000] "GET /tareac2/ HTTP/1.1" 401 809 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36"
192.168.1.176 - - [03/Nov/2023:11:32:40 +0000] "-" 408 0 "-" "-"
192.168.1.176 - gondavid [03/Nov/2023:11:32:58 +0000] "GET /tareac2/ HTTP/1.1" 200 821 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36"
192.168.1.176 - - [03/Nov/2023:11:33:50 +0000] "-" 408 0 "-" "-"
192.168.1.175 - - [03/Nov/2023:11:34:56 +0000] "GET /tareac2/ HTTP/1.1" 401 809 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36"
192.168.1.175 - gardavid [03/Nov/2023:11:35:23 +0000] "GET /tareac2/ HTTP/1.1" 200 821 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36"
192.168.1.175 - - [03/Nov/2023:11:35:23 +0000] "GET /favicon.ico HTTP/1.1" 404 491 "http://192.168.1.205/tareac2/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36"
192.168.1.175 - - [03/Nov/2023:11:36:15 +0000] "-" 408 0 "-" "-"
servidordgg@servidordgg:~$ _
```

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

F) Módulos status e info

status e info son módulos de monitorización. En concreto:

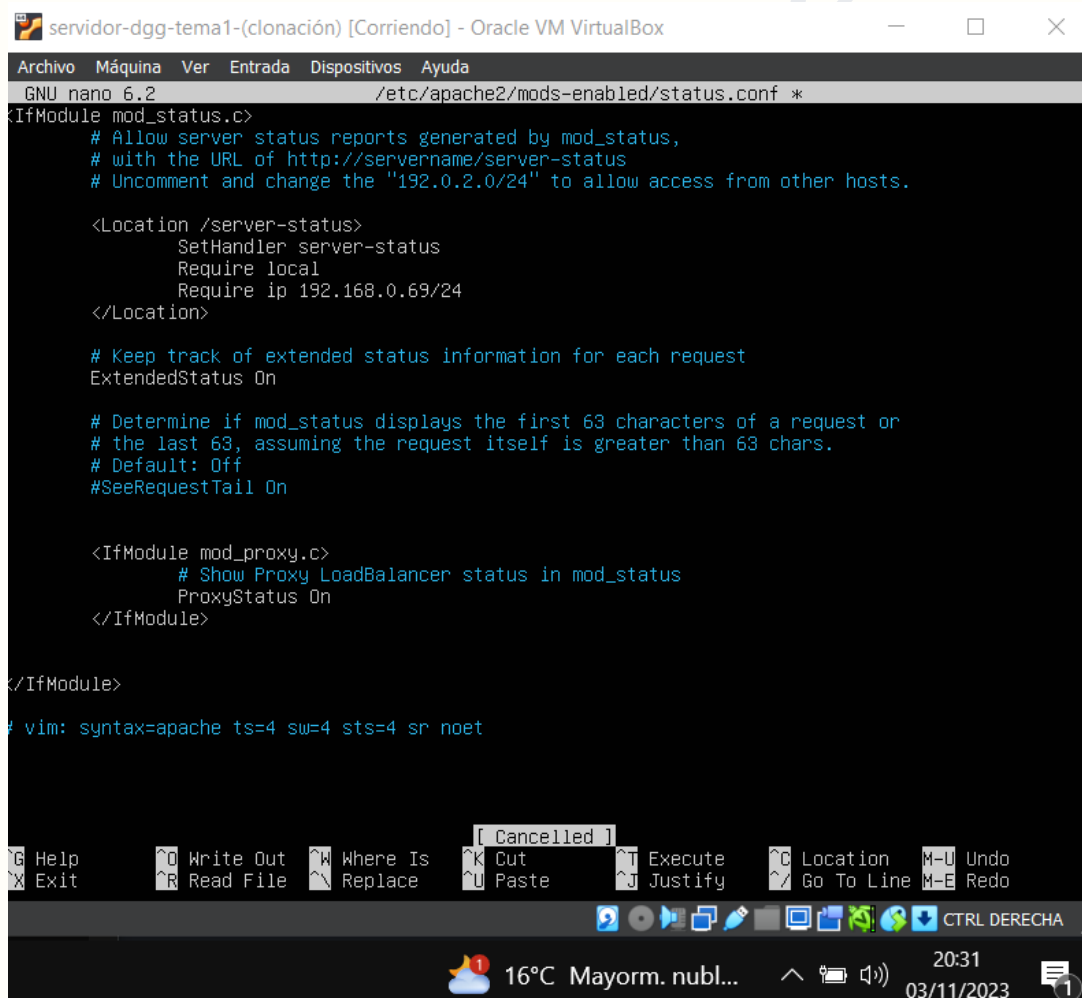
status permite monitorizar el rendimiento del servidor Apache (generando un HTML).
info proporciona una vista resumida de la configuración del servidor.

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

Para ello usaremos:

```
$ a2enmod status
```

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



```
servidor-dgg-tema1-(clonación) [Corriendo] - Oracle VM VirtualBox
GNU nano 6.2 /etc/apache2/mods-enabled/status.conf *
<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 192.168.0.69/24
</Location>

# Keep track of extended status information for each request
ExtendedStatus On

# Determine if mod_status displays the first 63 characters of a request or
# the last 63, assuming the request itself is greater than 63 chars.
# Default: Off
#SeeRequestTail On

<IfModule mod_proxy.c>
    # Show Proxy LoadBalancer status in mod_status
    ProxyStatus On
</IfModule>

</IfModule>

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

PASO 3) Reinicia el servidor para aplicar los cambios.

Para ello usaremos:

```
$ systemctl restart apache2
```

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

Apache Server Status for 192.168.0.205 (via 192.168.0.205)

Server Version: Apache/2.4.52 (Ubuntu)
Server MPM: event
Server Built: 2023-05-03T20:02:51

Current Time: Friday, 03-Nov-2023 19:54:45 UTC
Restart Time: Friday, 03-Nov-2023 19:54:22 UTC
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 23 seconds
Server load: 0.04 0.33 0.21
Total accesses: 1 - Total Traffic: 0 kB - Total Duration: 0
CPU Usage: u0 s.07 cu0 cs0 - .304% CPU load
.0435 requests/sec - 0 B/second - 0 B/request - 0 ms/request
1 requests currently being processed, 49 idle workers

Slot	PID	Stopping	Connections			Threads			Async connections		
			total	accepting	busy	idle	writing	keep-alive	closing		
0	1226	no	0	yes	0	25	0	0	0	0	
1	1227	no	0	yes	1	24	0	0	0	0	
Sum	2	0	0		1	49	0	0	0	0	

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,
"T" 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	1226	0/1/1	.	0.07	13	0	0	0.00	0.00	0.00	192.168.0.69	http/1.1	127.0.1.1:80	GET /server-info HTTP/1.1
1-0	1227	1/0/0	W	0.00	0	0	0	0.00	0.00	0.00	192.168.0.69	http/1.1	127.0.1.1:80	GET /server-status HTTP/1.1

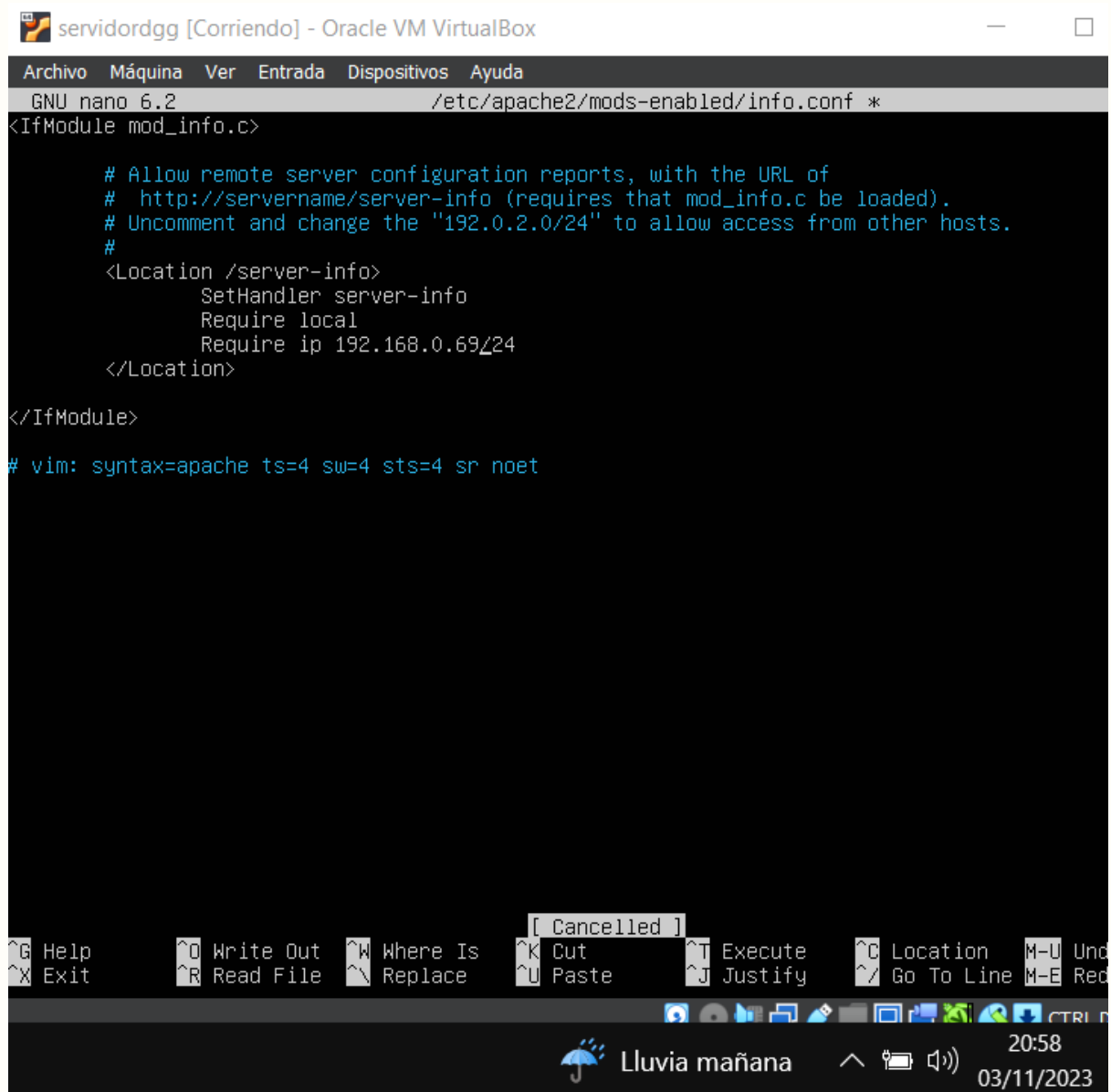
Srv Child Server number - generation
PID OS process ID
Acc Number of accesses this connection / this child / this slot

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

Para ello usaremos:

```
$ a2enmod info
```

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



```
servidordgg [Corriendo] - Oracle VM VirtualBox
GNU nano 6.2 /etc/apache2/mods-enabled/info.conf *
<IfModule mod_info.c>

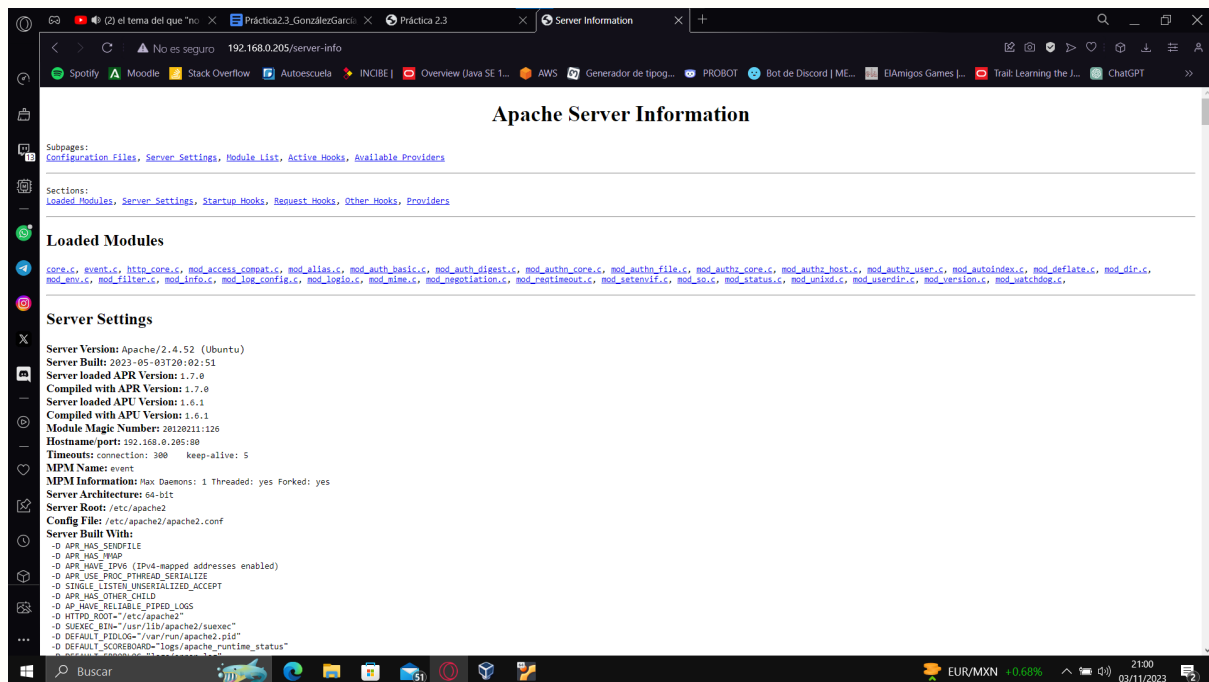
# Allow remote server configuration reports, with the URL of
# http://servername/server-info (requires that mod_info.c be loaded).
# Uncomment and change the "192.0.2.0/24" to allow access from other hosts.
#
<Location /server-info>
    SetHandler server-info
    Require local
    Require ip 192.168.0.69/24
</Location>
</IfModule>

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

PASO 7) Reinicia el servidor para aplicar los cambios.

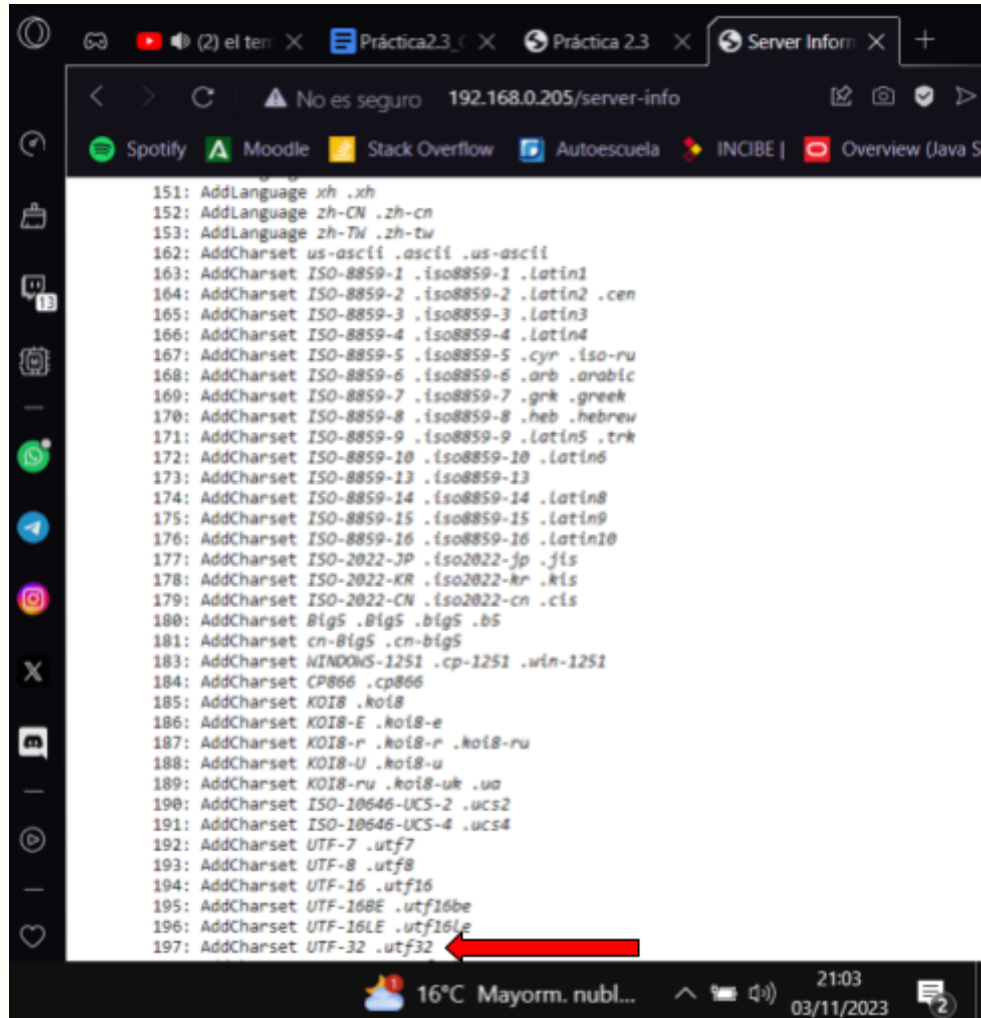
Para ello usaremos:

```
$ systemctl restart apache2
```

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?

Si, está activado y además carga la configuración UTF-32



```
151: AddLanguage xh .xh
152: AddLanguage zh-CN .zh-cn
153: AddLanguage zh-TW .zh-tw
162: AddCharset us-ascii .ascii .us-ascii
163: AddCharset ISO-8859-1 .iso8859-1 .latin1
164: AddCharset ISO-8859-2 .iso8859-2 .latin2 .cen
165: AddCharset ISO-8859-3 .iso8859-3 .latin3
166: AddCharset ISO-8859-4 .iso8859-4 .latin4
167: AddCharset ISO-8859-5 .iso8859-5 .cyr .iso-ru
168: AddCharset ISO-8859-6 .iso8859-6 .arb .arabic
169: AddCharset ISO-8859-7 .iso8859-7 .grk .greek
170: AddCharset ISO-8859-8 .iso8859-8 .heb .hebrew
171: AddCharset ISO-8859-9 .iso8859-9 .latin5 .trk
172: AddCharset ISO-8859-10 .iso8859-10 .latin6
173: AddCharset ISO-8859-13 .iso8859-13
174: AddCharset ISO-8859-14 .iso8859-14 .latin8
175: AddCharset ISO-8859-15 .iso8859-15 .latin9
176: AddCharset ISO-8859-16 .iso8859-16 .latin10
177: AddCharset ISO-2022-JP .iso2022-jp .jis
178: AddCharset ISO-2022-KR .iso2022-kr .kis
179: AddCharset ISO-2022-CN .iso2022-cn .cis
180: AddCharset Big5 .Big5 .big5 .b5
181: AddCharset cn-Big5 .cn-big5
183: AddCharset WINDOWS-1251 .cp-1251 .win-1251
184: AddCharset CP866 .cp866
185: AddCharset KOI8 .koi8
186: AddCharset KOI8-E .koi8-e
187: AddCharset KOI8-R .koi8-r .koi8-ru
188: AddCharset KOI8-U .koi8-u
189: AddCharset KOI8-RU .koi8-uk .ua
190: AddCharset ISO-10646-UCS-2 .ucs2
191: AddCharset ISO-10646-UCS-4 .ucs4
192: AddCharset UTF-7 .utf7
193: AddCharset UTF-8 .utf8
194: AddCharset UTF-16 .utf16
195: AddCharset UTF-16BE .utf16be
196: AddCharset UTF-16LE .utf16le
197: AddCharset UTF-32 .utf32
```

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.

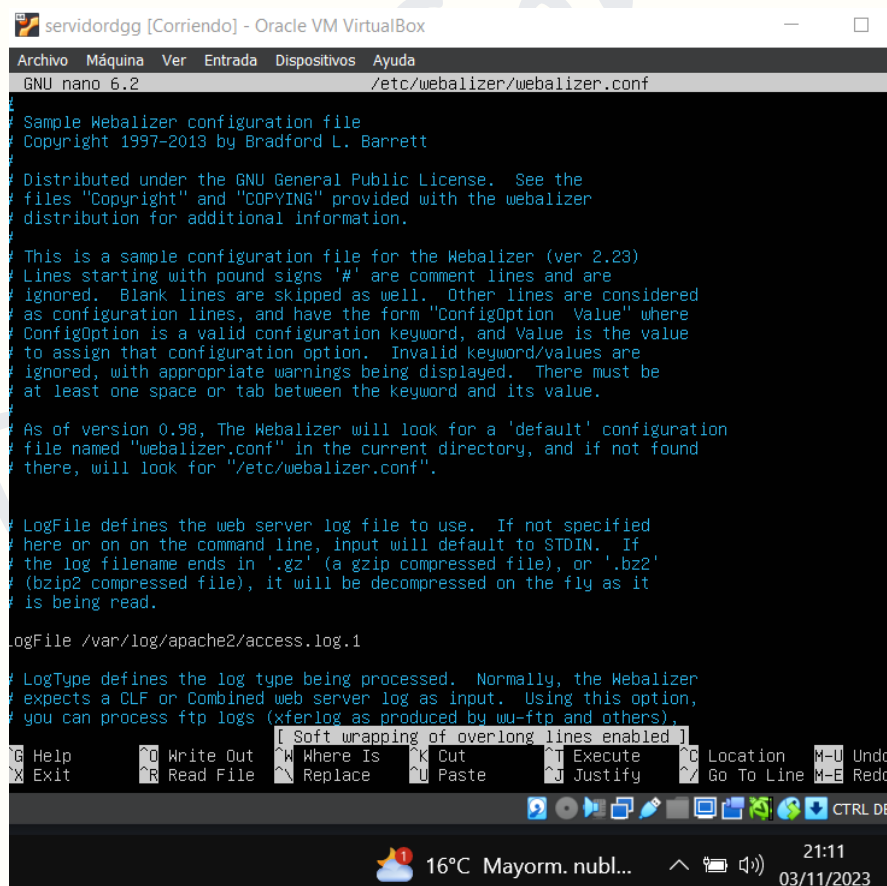
PASO 1) En tu servidor Linux, instala la aplicación Webalizer (usa apt-get install, pero antes actualiza el servidor Linux).

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?

Los coge de: /var/log/apache2/access.log.1

¿Es correcta la ruta y el nombre del fichero? Si no es así, modifícala.

Si, es correcta



```
servidordgg [Corriendo] - Oracle VM VirtualBox
GNU nano 6.2 /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),

[ Soft wrapping of overlong lines enabled ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   ^M-U Undo
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^J Justify    ^_ Go To Line  ^M-E Redo
CTRL DE

16°C Mayorm. nubl... 21:11 03/11/2023
```

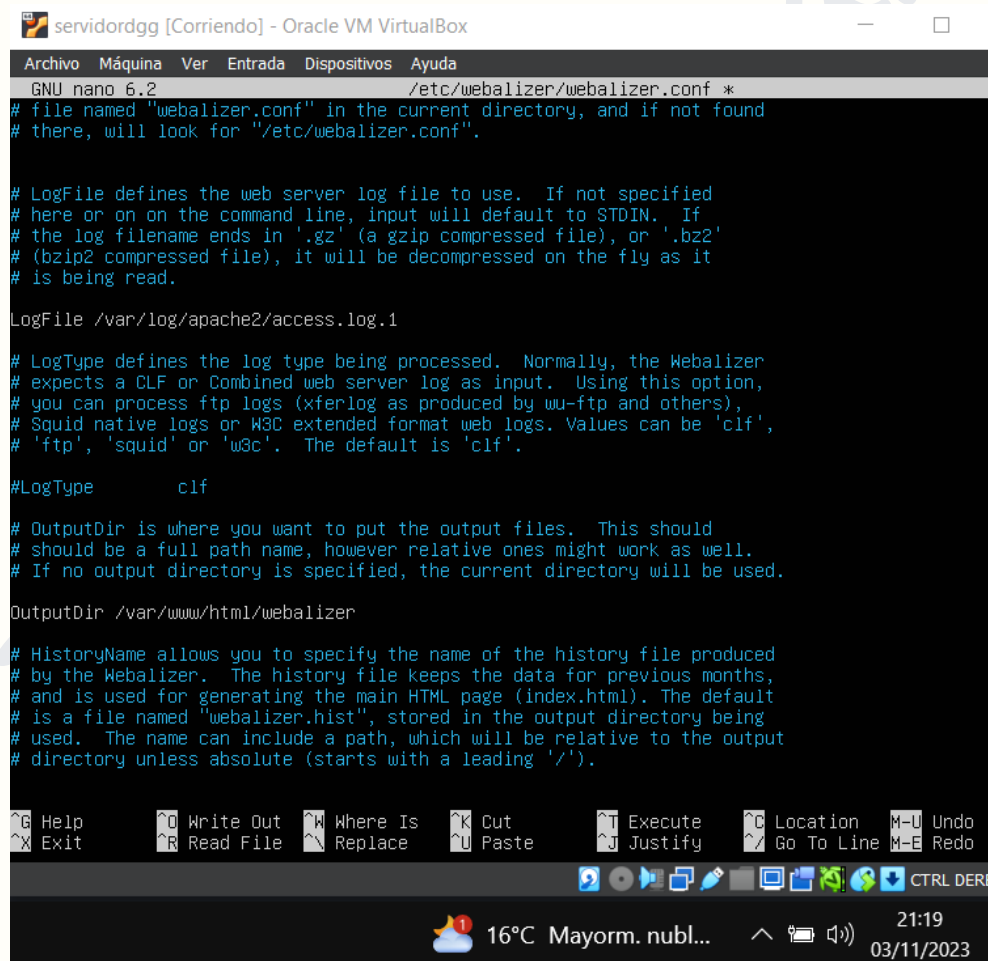
PASO 3) La instalación también implica la creación del recurso que se servirá desde el navegador, ¿Dónde está este fichero? ¿Es correcta la ubicación para servirlo? Si no es así, muévelo a la ubicación correcta.

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
```



```
servidordgg [Corriendo] - Oracle VM VirtualBox
GNU nano 6.2 /etc/webalizer/webalizer.conf *
# 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 '/').

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

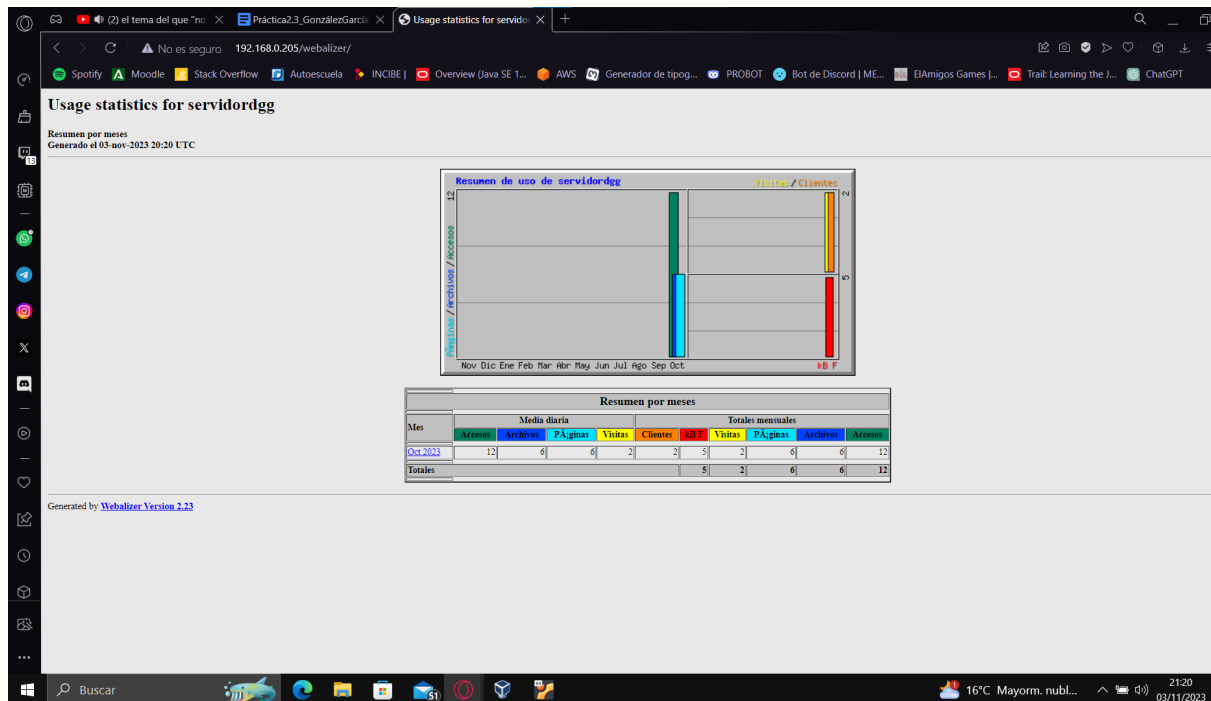
CTRL DEB...
```


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.

Para ello usaremos:

```
$ sudo webalizer
```

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



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.

No puedo debido a los token.

David González