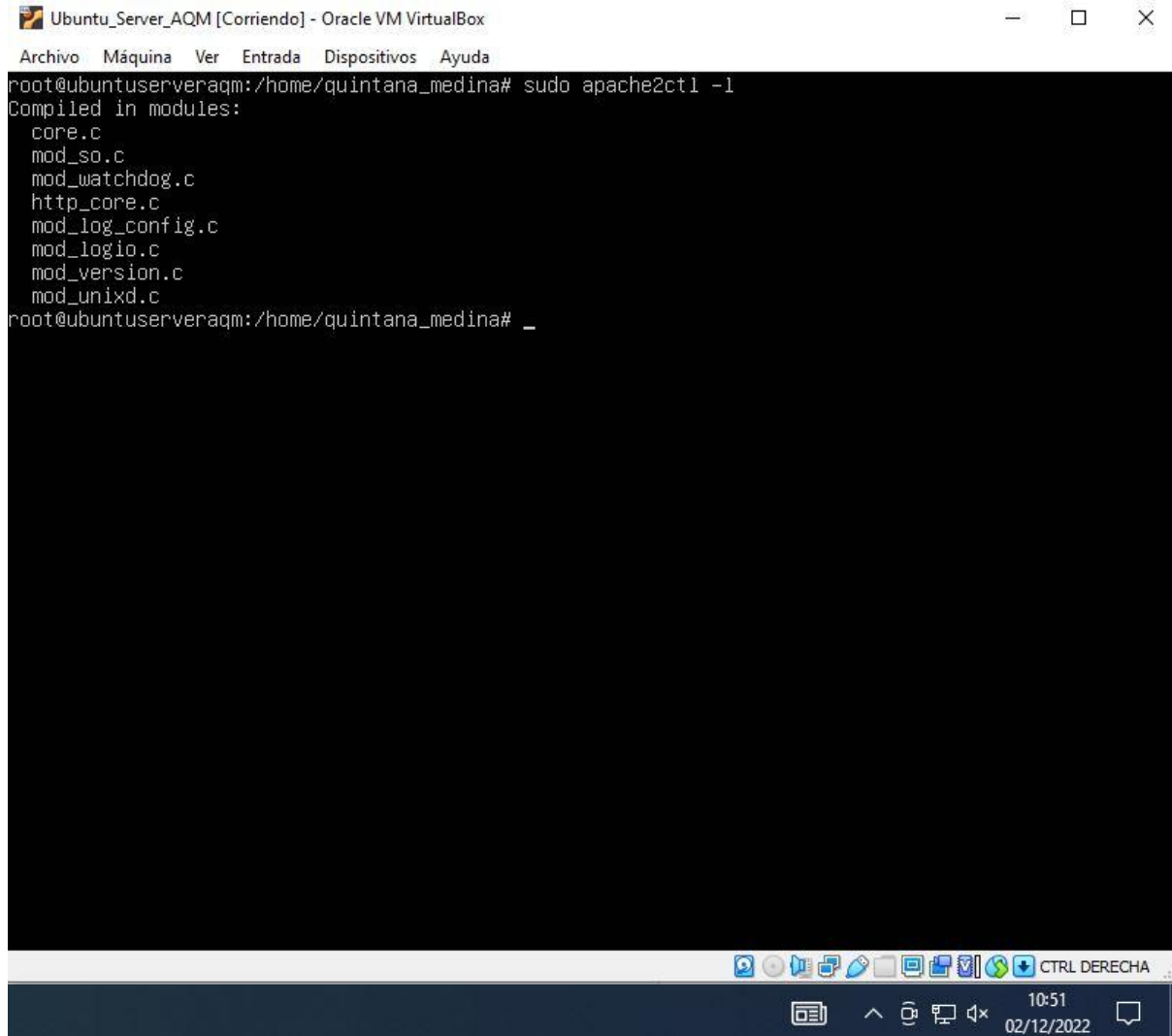


Contraseña: alumno(inicial nombre y apellido)

A.1) Módulos

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



```
Ubuntu_Server_AQM [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
root@ubuntuserveraqm:/home/quintana_medina# sudo apache2ctl -l
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
root@ubuntuserveraqm:/home/quintana_medina# _
```

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

Haciendo uso del comando `ls` hemos listado todos los elementos de la carpeta mencionada un poco más arriba, pudiendo comprobar los módulos requeridos.

```

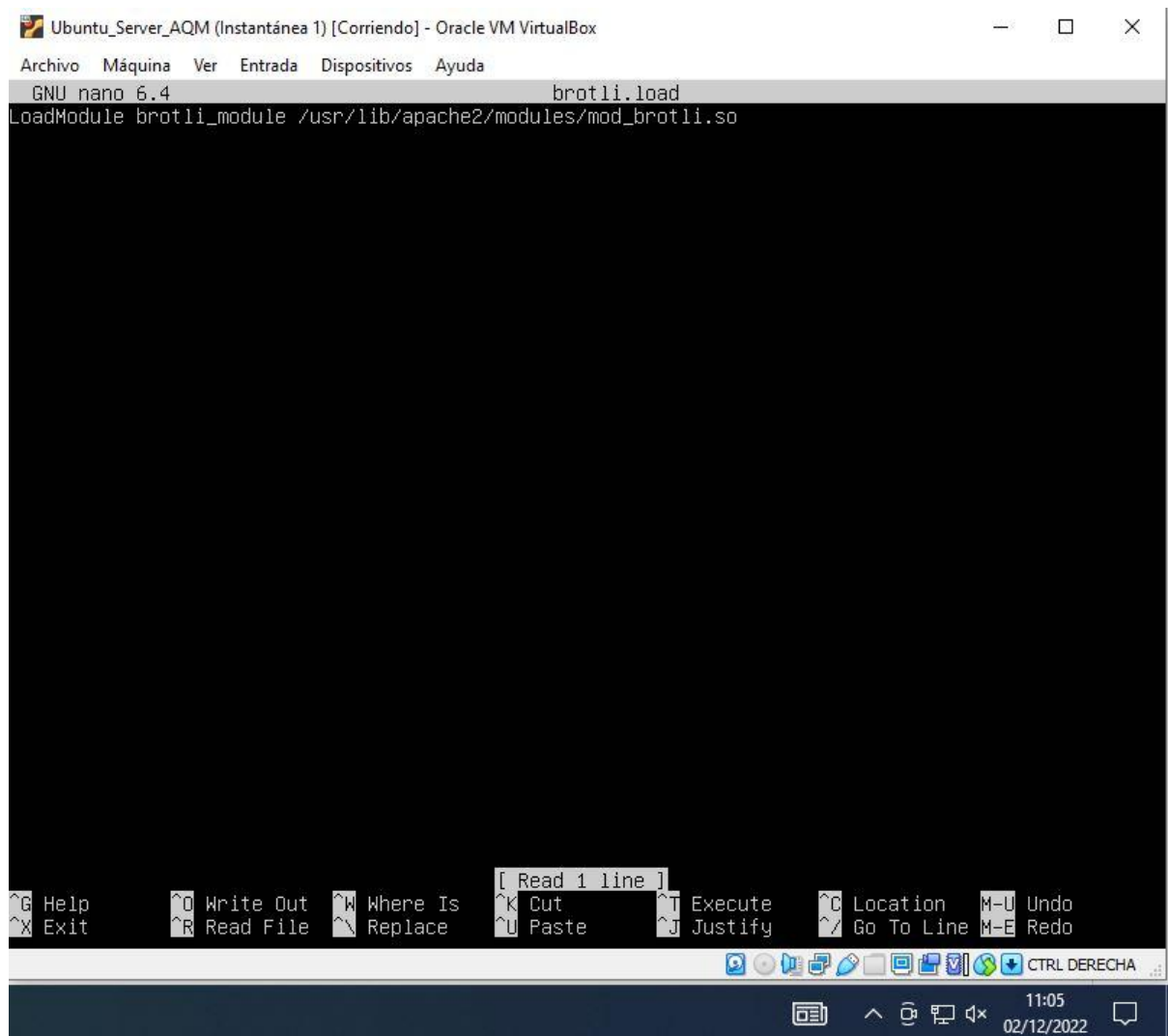
Ubuntu_Server_AQM [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
access_compat.load  cgid.load  log_debug.load  ratelimit.load
actions.conf  charset_lite.load  log_forensic.load  reflector.load
actions.load  data.load  lua.load  remoteip.load
alias.conf  dav.load  macro.load  reqtimeout.conf
alias.load  dav_fs.conf  md.load  reqtimeout.load
allowmethods.load  dav_fs.load  mime.conf  request.load
asis.load  dav_lock.load  mime.load  rewrite.load
auth_basic.load  dbd.load  mime_magic.conf  sed.load
auth_digest.load  deflate.conf  mime_magic.load  session.load
auth_form.load  deflate.load  mpm_event.conf  session_cookie.load
authn_anon.load  dialup.load  mpm_event.load  session_crypto.load
authn_core.load  dir.conf  mpm_prefork.conf  session_dbd.load
authn_dbd.load  dir.load  mpm_prefork.load  setenvif.conf
authn_dbm.load  dump_io.load  mpm_worker.conf  setenvif.load
authn_file.load  echo.load  mpm_worker.load  slotmem_plain.load
authn_socache.load  env.load  negotiation.conf  slotmem_shm.load
authnz_fcgi.load  expires.load  negotiation.load  socache_dbm.load
authnz_ldap.load  ext_filter.load  proxy.conf  socache_memcache.load
authz_core.load  file_cache.load  proxy.load  socache_redis.load
authz_dbd.load  filter.load  proxy_ajp.load  socache_shmcb.load
authz_dbm.load  headers.load  proxy_balancer.conf  spelling.load
authz_groupfile.load  heartbeat.load  proxy_balancer.load  ssl.conf
authz_host.load  heartmonitor.load  proxy_connect.load  ssl.load
authz_owner.load  http2.conf  proxy_express.load  status.conf
authz_user.load  http2.load  proxy_fcgi.load  status.load
autoindex.conf  ident.load  proxy_fdpass.load  substitute.load
autoindex.load  imagemap.load  proxy_ftp.conf  suexec.load
brotli.load  include.load  proxy_ftp.load  unique_id.load
buffer.load  info.conf  proxy_hcheck.load  userdir.conf
cache.load  info.load  proxy_html.conf  userdir.load
cache_disk.conf  lbmethod_bybusyness.load  proxy_html.load  usertrack.load
cache_disk.load  lbmethod_byrequests.load  proxy_http.load  vhost_alias.load
cache_socache.load  lbmethod_bytraffic.load  proxy_http2.load  xml2enc.load
cern_meta.load  lbmethod_heartbeat.load  proxy_scgi.load
cgi.load  ldap.conf  proxy_uwsgi.load
cgid.conf  ldap.load  proxy_wstunnel.load
root@ubuntu:~#

```

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?

Hemos usado el editor nano para comprobar el contenido de uno de los archivos load. Este viene “dividido” en varias partes, dado que el primero indica el tipo de directiva (LoadModule), luego aparece el módulo en sí y por último el nombre del archivo, sirviendo todo ello para cargar, en este caso, el módulo dinámico indicado.

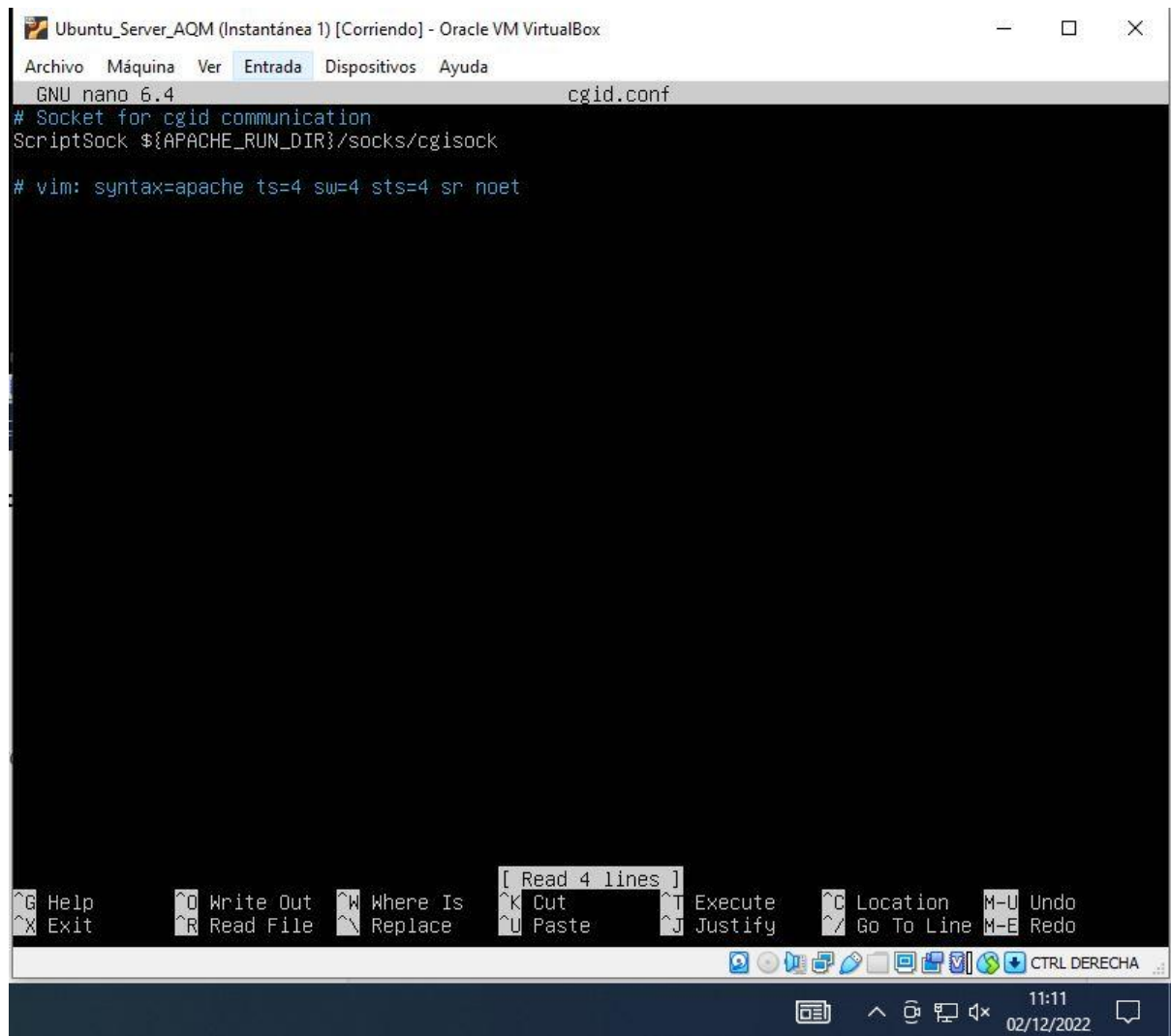
Su extensión, como puede verse más abajo, es “.so”.



The screenshot shows a terminal window titled "Ubuntu_Server_AQM (Instantánea 1) [Corriendo] - Oracle VM VirtualBox". The terminal is running the GNU nano 6.4 text editor, editing the file "brotli.load". The first line of the file contains the command "LoadModule brotli_module /usr/lib/apache2/modules/mod_brotli.so". The nano editor's status bar at the bottom shows various keyboard shortcuts: ^G Help, ^X Exit, ^O Write Out, ^R Read File, ^W Where Is, ^_ Replace, [Read 1 line], ^K Cut, ^U Paste, ^T Execute, ^J Justify, ^C Location, ^_ Go To Line, ^M-U Undo, and ^M-E Redo. The bottom of the window shows a taskbar with system icons and the date/time "11:05 02/12/2022".

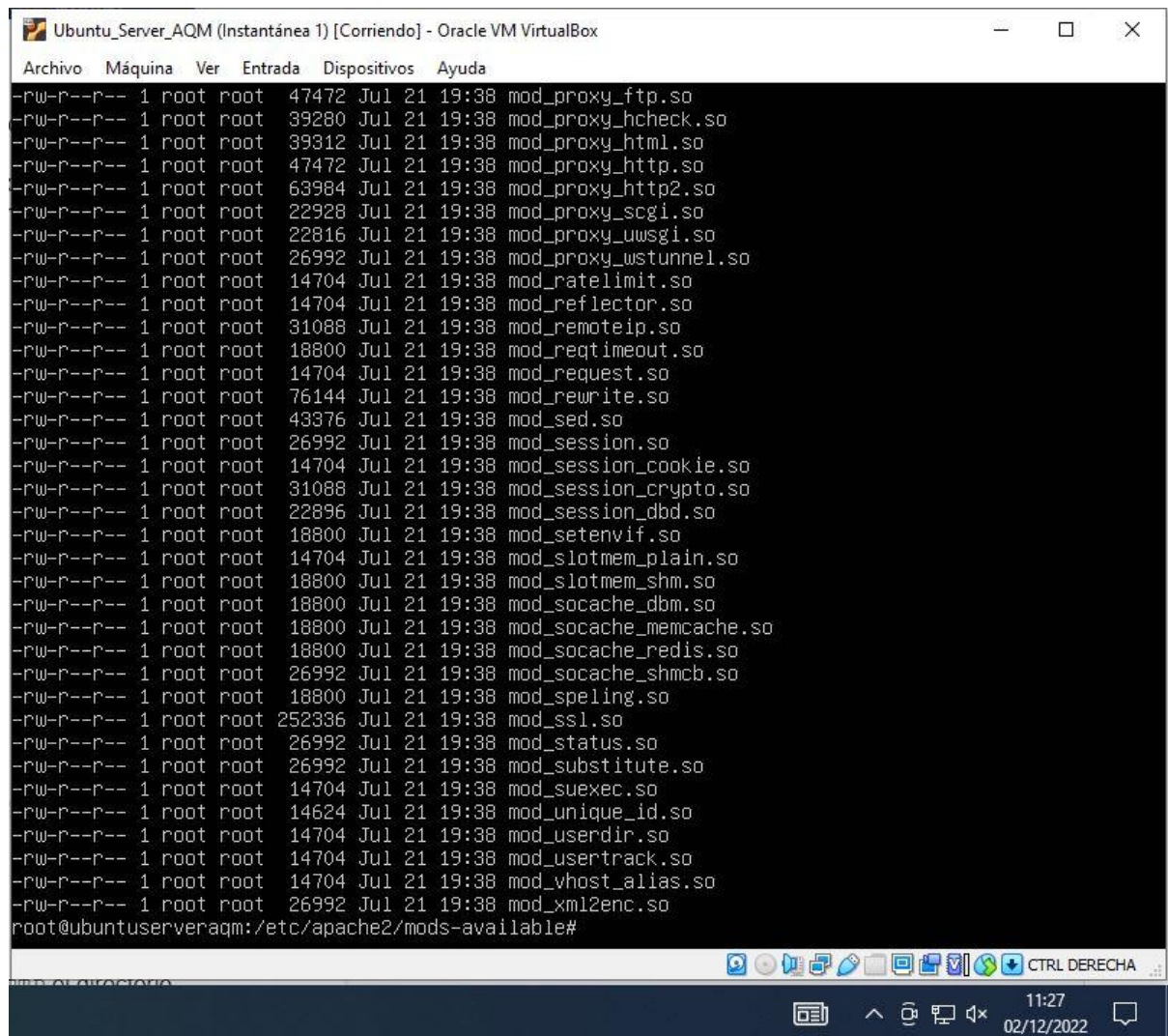
PASO 4) Edita uno de los archivos `.conf` y observa cómo se añaden directivas dentro del módulo.¿Qué etiquetas se utilizan en estos archivos?

Mismo procedimiento de la vez anterior, usamos cat y comprobamos su contenido. Podemos comprobar la directiva apache y cómo se aplican en sockets.

A screenshot of a virtual machine window titled 'Ubuntu_Server_AQM (Instantánea 1) [Corriendo] - Oracle VM VirtualBox'. Inside the VM, the nano 6.4 text editor is open, editing a file named 'cgid.conf'. The editor's menu bar includes 'Archivo', 'Máquina', 'Ver', 'Entrada', 'Dispositivos', and 'Ayuda'. The file content shows a comment '# Socket for cgid communication', a 'ScriptSock' directive pointing to '\$APACHE_RUN_DIR/socks/cgisock', and a vim configuration line. The status bar at the bottom of the editor shows '[Read 4 lines]' and various keyboard shortcuts like '^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', and '^M-E Redo'. The bottom of the screen shows a taskbar with system icons and a clock displaying '11:11' on '02/12/2022'.

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

Consultamos el archivo con `ls -l`, pudiendo comprobar que contiene los módulos de apache, pero compilados de forma individual con la extensión “.so” (la cual vimos antes).



```
Ubuntu_Server_AQM (Instantánea 1) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
-rw-r--r--  1 root root  47472 Jul 21 19:38 mod_proxy_ftp.so
-rw-r--r--  1 root root  39280 Jul 21 19:38 mod_proxy_hcheck.so
-rw-r--r--  1 root root  39312 Jul 21 19:38 mod_proxy_html.so
-rw-r--r--  1 root root  47472 Jul 21 19:38 mod_proxy_http.so
-rw-r--r--  1 root root  63984 Jul 21 19:38 mod_proxy_http2.so
-rw-r--r--  1 root root  22928 Jul 21 19:38 mod_proxy_scgi.so
-rw-r--r--  1 root root  22816 Jul 21 19:38 mod_proxy_uwsgi.so
-rw-r--r--  1 root root  26992 Jul 21 19:38 mod_proxy_wstunnel.so
-rw-r--r--  1 root root  14704 Jul 21 19:38 mod_ratelimit.so
-rw-r--r--  1 root root  14704 Jul 21 19:38 mod_reflector.so
-rw-r--r--  1 root root  31088 Jul 21 19:38 mod_remoteip.so
-rw-r--r--  1 root root  18800 Jul 21 19:38 mod_reqtimeout.so
-rw-r--r--  1 root root  14704 Jul 21 19:38 mod_request.so
-rw-r--r--  1 root root  76144 Jul 21 19:38 mod_rewrite.so
-rw-r--r--  1 root root  43376 Jul 21 19:38 mod_sed.so
-rw-r--r--  1 root root  26992 Jul 21 19:38 mod_session.so
-rw-r--r--  1 root root  14704 Jul 21 19:38 mod_session_cookie.so
-rw-r--r--  1 root root  31088 Jul 21 19:38 mod_session_crypto.so
-rw-r--r--  1 root root  22896 Jul 21 19:38 mod_session_dbd.so
-rw-r--r--  1 root root  18800 Jul 21 19:38 mod_setenvif.so
-rw-r--r--  1 root root  14704 Jul 21 19:38 mod_slotmem_plain.so
-rw-r--r--  1 root root  18800 Jul 21 19:38 mod_slotmem_shm.so
-rw-r--r--  1 root root  18800 Jul 21 19:38 mod_socache_dbm.so
-rw-r--r--  1 root root  18800 Jul 21 19:38 mod_socache_memcache.so
-rw-r--r--  1 root root  18800 Jul 21 19:38 mod_socache_redis.so
-rw-r--r--  1 root root  26992 Jul 21 19:38 mod_socache_shmcb.so
-rw-r--r--  1 root root  18800 Jul 21 19:38 mod_speling.so
-rw-r--r--  1 root root 252336 Jul 21 19:38 mod_ssl.so
-rw-r--r--  1 root root  26992 Jul 21 19:38 mod_status.so
-rw-r--r--  1 root root  26992 Jul 21 19:38 mod_substitute.so
-rw-r--r--  1 root root  14704 Jul 21 19:38 mod_suexec.so
-rw-r--r--  1 root root  14624 Jul 21 19:38 mod_unique_id.so
-rw-r--r--  1 root root  14704 Jul 21 19:38 mod_userdir.so
-rw-r--r--  1 root root  14704 Jul 21 19:38 mod_usertrack.so
-rw-r--r--  1 root root  14704 Jul 21 19:38 mod_vhost_alias.so
-rw-r--r--  1 root root  26992 Jul 21 19:38 mod_xml2enc.so
root@ubuntu-server-aqm:/etc/apache2/mods-available#
```

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

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/>

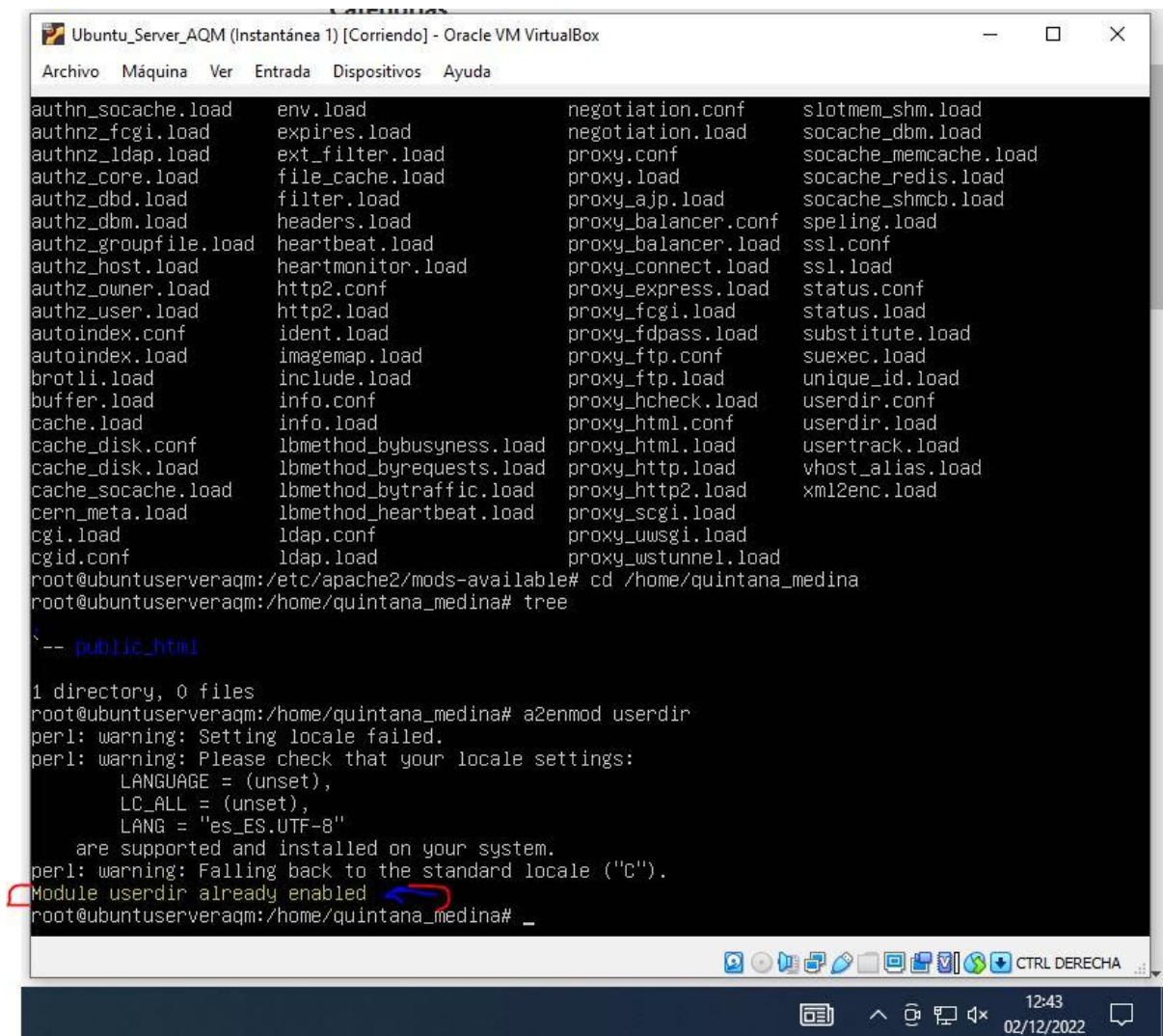
PASO 1) Comprueba si el módulo `userdir` está habilitado. ¿Lo está?

Hemos comprobado si se encontraba en el listado y no aparecía, así que seguimos varios pasos para poder habilitarlo. haciendo uso del comando `"a2enmod userdir"`, además de crear el directorio correspondiente en mi usuario.

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

PASO 3) Verifica ahora si el módulo está habilitado.

Aquí podemos ver como indica su activación.



```
Ubuntu_Server_AQM (Instantánea 1) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

authn_socache.load  env.load  negotiation.conf  slotmem_shm.load
authnz_fcgi.load    expires.load  negotiation.load  socache_dbm.load
authnz_ldap.load    ext_filter.load  proxy.conf        socache_memcache.load
authz_core.load     file_cache.load  proxy.load        socache_redis.load
authz_dbd.load      filter.load     proxy_ajp.load    socache_shmcb.load
authz_dbm.load      headers.load    proxy_balancer.conf  spelling.load
authz_groupfile.load  heartbeat.load  proxy_balancer.load  ssl.conf
authz_host.load     heartmonitor.load  proxy_connect.load  ssl.load
authz_owner.load    http2.conf      proxy_express.load  status.conf
authz_user.load     http2.load      proxy_fcgi.load     status.load
autoindex.conf      ident.load      proxy_fdpass.load   substitute.load
autoindex.load      imagemap.load  proxy_ftp.conf      suexec.load
brotli.load         include.load    proxy_ftp.load      unique_id.load
buffer.load         info.conf      proxy_hcheck.load   userdir.conf
cache.load          info.load      proxy_html.conf     userdir.load
cache_disk.conf     lbmethod_bybusyness.load  proxy_html.load    usertrack.load
cache_disk.load     lbmethod_byrequests.load  proxy_http.load     vhost_alias.load
cache_socache.load  lbmethod_bytraffic.load  proxy_http2.load    xml2enc.load
cern_meta.load      ldap.conf      proxy_scgi.load
cgi.load            ldap.load      proxy_uwsgi.load
cgid.conf           ldap.load      proxy_wstunnel.load

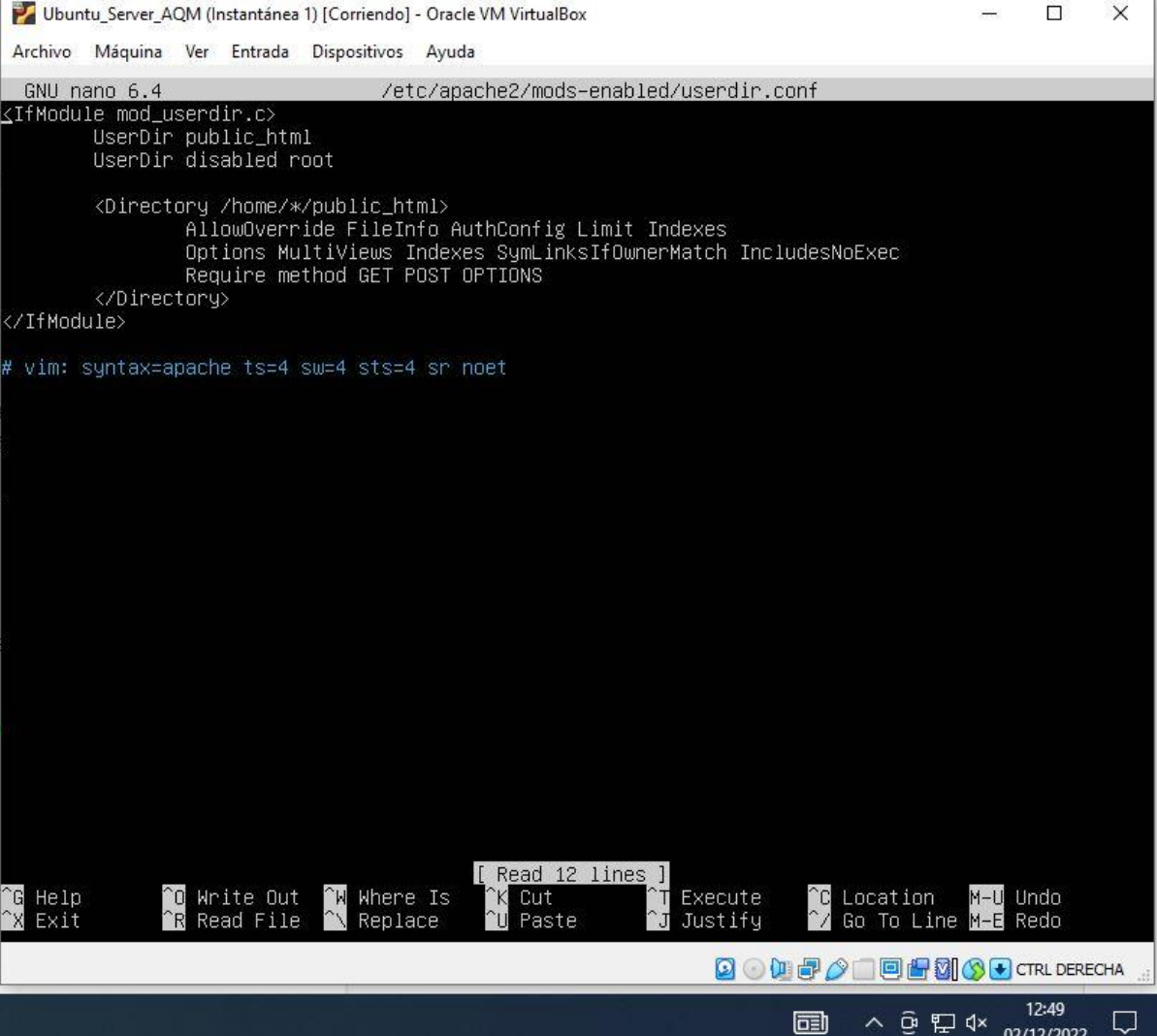
root@ubuntuserveraqm:/etc/apache2/mods-available# cd /home/quintana_medina
root@ubuntuserveraqm:/home/quintana_medina# tree
.
-- public_html

1 directory, 0 files
root@ubuntuserveraqm:/home/quintana_medina# a2enmod userdir
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
    LANGUAGE = (unset),
    LC_ALL = (unset),
    LANG = "es_ES.UTF-8"
    are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").
Module userdir already enabled
root@ubuntuserveraqm:/home/quintana_medina#
```

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?

Según vemos comprobando el archivo en cuestión, sólo el usuario root se encuentra deshabilitado. Por otro lado, para las páginas personales los usuarios deben crear “public_html”. Tras ello, no deben disponer de mayor dificultad.



The screenshot shows a VirtualBox window titled "Ubuntu_Server_AQM (Instantánea 1) [Corriendo] - Oracle VM VirtualBox". Inside, the GNU nano 6.4 text editor is open, editing the file /etc/apache2/mods-enabled/userdir.conf. The file content is as follows:

```
<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>
```

At the bottom of the editor, a status line reads: "# vim: syntax=apache ts=4 sw=4 sts=4 sr noet". The bottom of the window shows a menu bar with various shortcuts (e.g., ^G Help, ^X Exit, ^O Write Out, ^R Read File, ^W Where Is, ^L Replace, ^K Cut, ^U Paste, ^T Execute, ^J Justify, ^C Location, ^_ Go To Line, M-U Undo, M-E Redo) and a system tray with icons for network, volume, and date/time (12:49, 02/12/2022).

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.

Tal y como se ha requerido, hemos abierto la página de usuario poniendo ~quintana_medina tras la ip del servidor.



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 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

Detalla los pasos seguidos para conseguirlo.

Por falta de pericia y velocidad, me ha sido imposible poder hacer esto en clase.

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?

Haciendo uso del comando “`apache2ctl -M`” podemos comprobar como, efectivamente, `authz_host` 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.

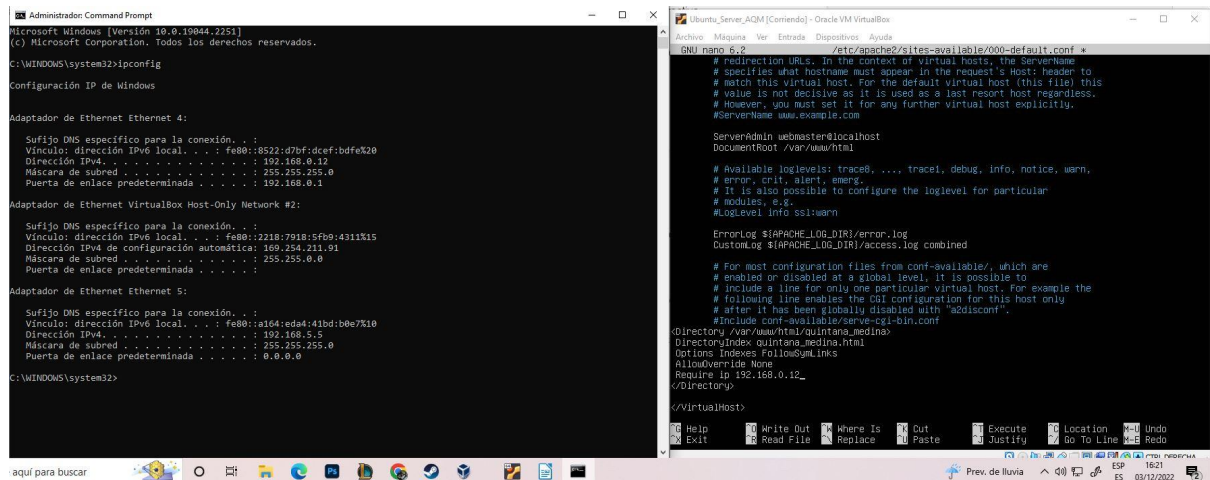
PASO 2) Edita el fichero de configuración

`/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).

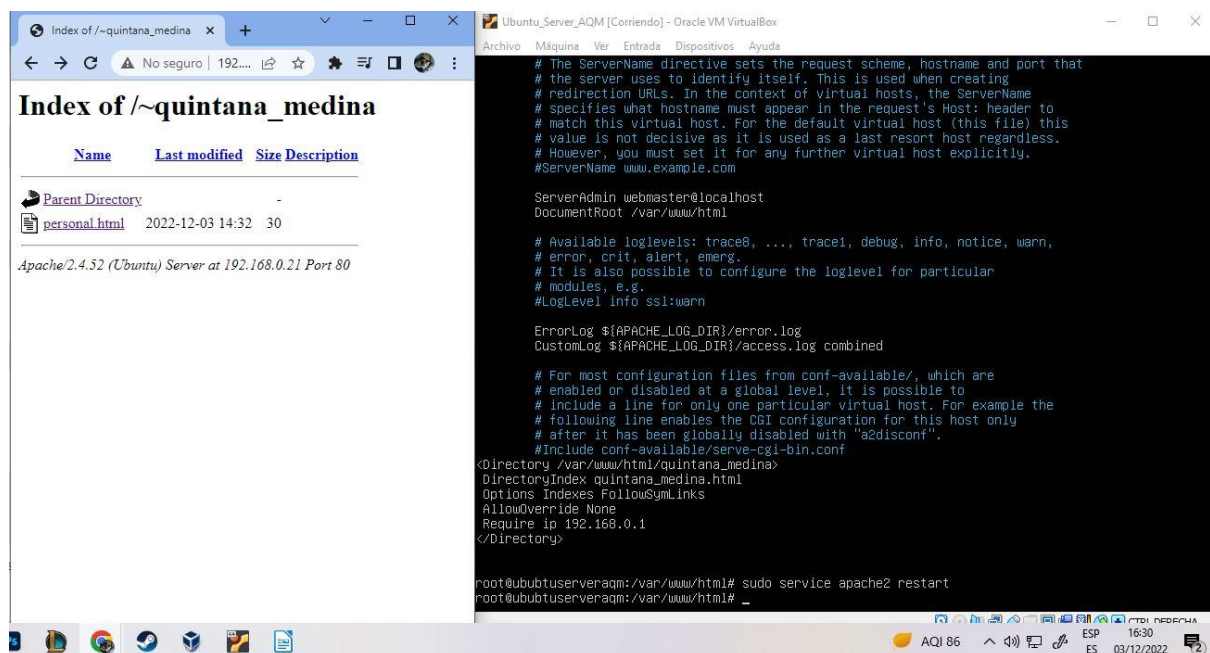
De acuerdo con la información dada en el enlace anterior, el uso de authz_host se puede emplear, tal y como indica el ejercicio, en el Directory indicado. En él, si queremos realizar la operación requerida basta con implementar en la etiqueta "require" las ips necesarias para poder permitir el acceso a este recurso.



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.

Juntamos aquí los recursos del paso 4 y 5.



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.

Desde la máquina del compañero:

Toma una captura de los pasos 7 y 9.

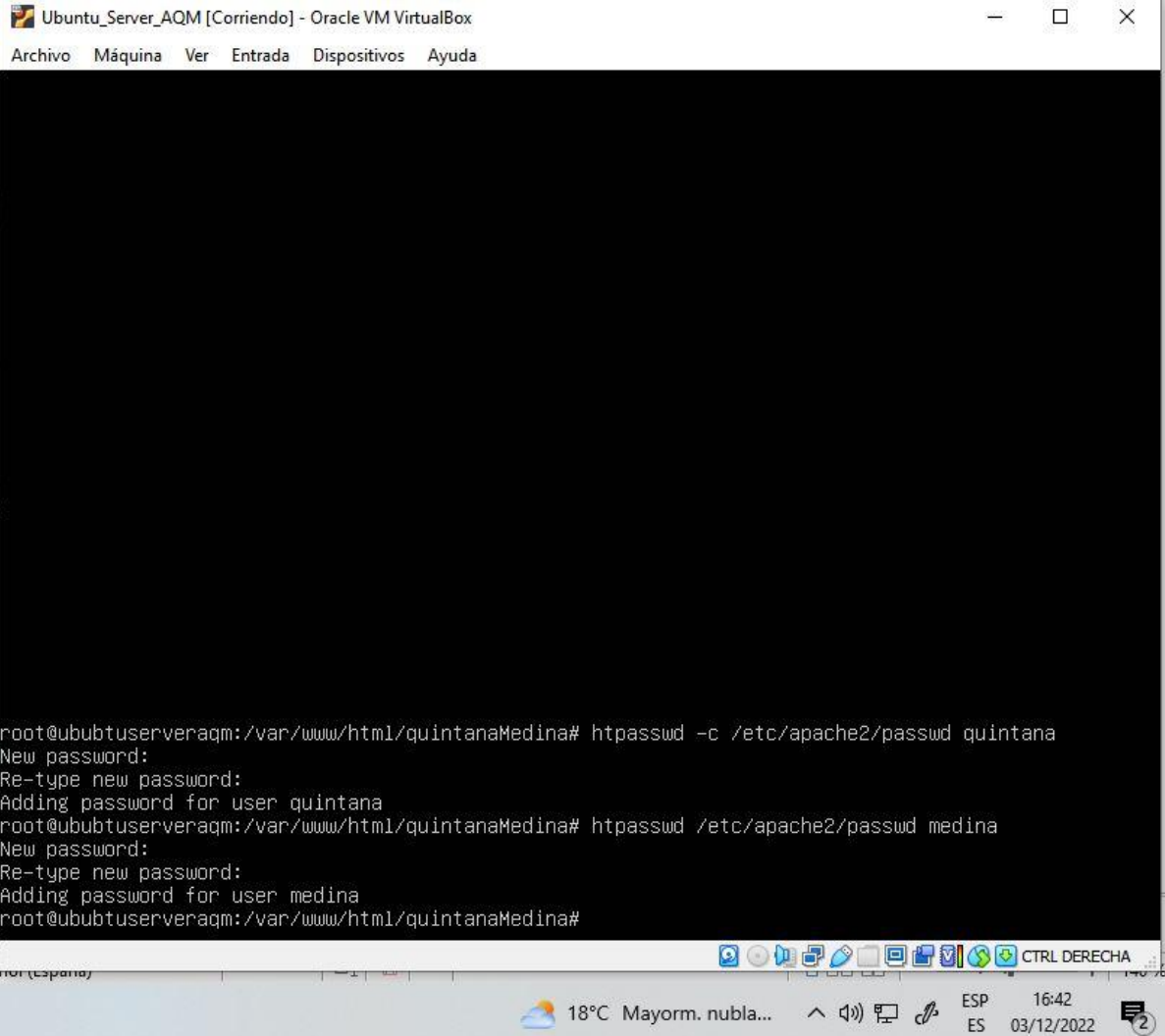
C.1) Autenticación Basic

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

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`.

Realizar esto ha resultado sencillo siguiendo los pasos indicados en el cuadro de trabajo: basta con usar el comando `htpasswd` y a continuación crear tanto el fichero como los usuarios, a los cuáles hemos dado la misma contraseña: `velazquez`.



Ubuntu_Server_AQM [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

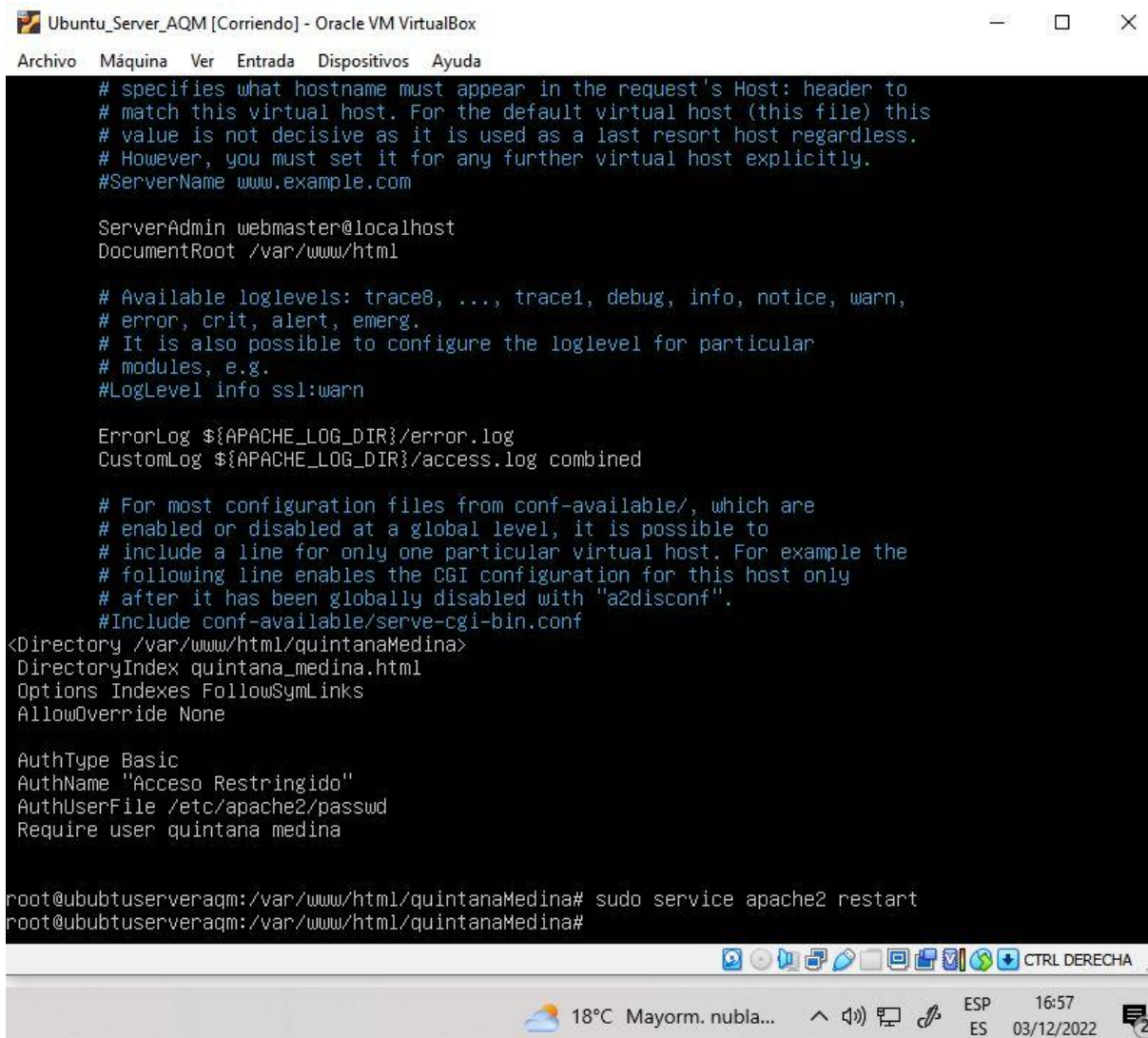
```
root@ububtuserveraqm:/var/www/html/quintanaMedina# htpasswd -c /etc/apache2/passwd quintana
New password:
Re-type new password:
Adding password for user quintana
root@ububtuserveraqm:/var/www/html/quintanaMedina# htpasswd /etc/apache2/passwd medina
New password:
Re-type new password:
Adding password for user medina
root@ububtuserveraqm:/var/www/html/quintanaMedina#
```

18°C Mayorm. nubla... ESP ES 16:42 03/12/2022

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).

Con nano añadimos las modificaciones necesarias



```
Ubuntu_Server_AQM [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

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

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

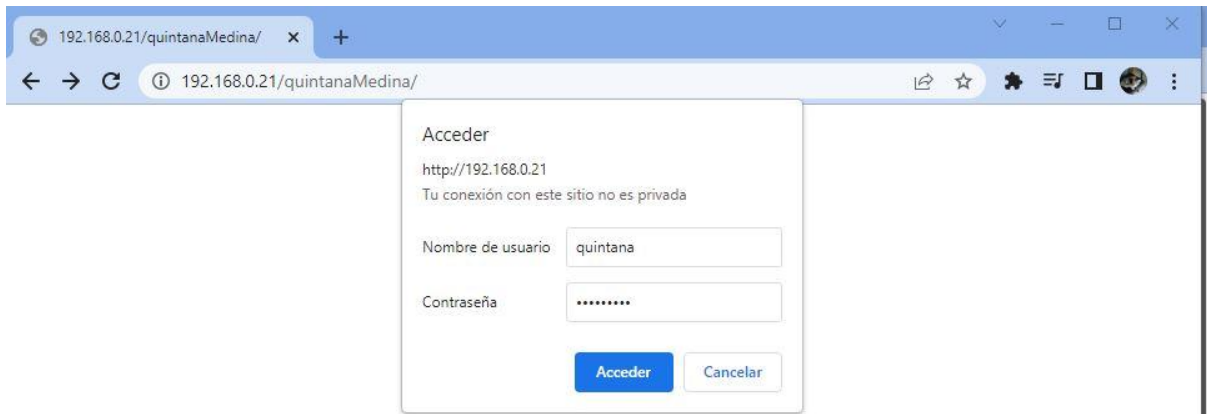
# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
<Directory /var/www/html/quintanaMedina>
  DirectoryIndex quintana_medina.html
  Options Indexes FollowSymLinks
  AllowOverride None

  AuthType Basic
  AuthName "Acceso Restringido"
  AuthUserFile /etc/apache2/passwd
  Require user quintana medina

root@ububtuserveraqm:/var/www/html/quintanaMedina# sudo service apache2 restart
root@ububtuserveraqm:/var/www/html/quintanaMedina#
```

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.



```
Ubuntu_Server_AQM [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

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

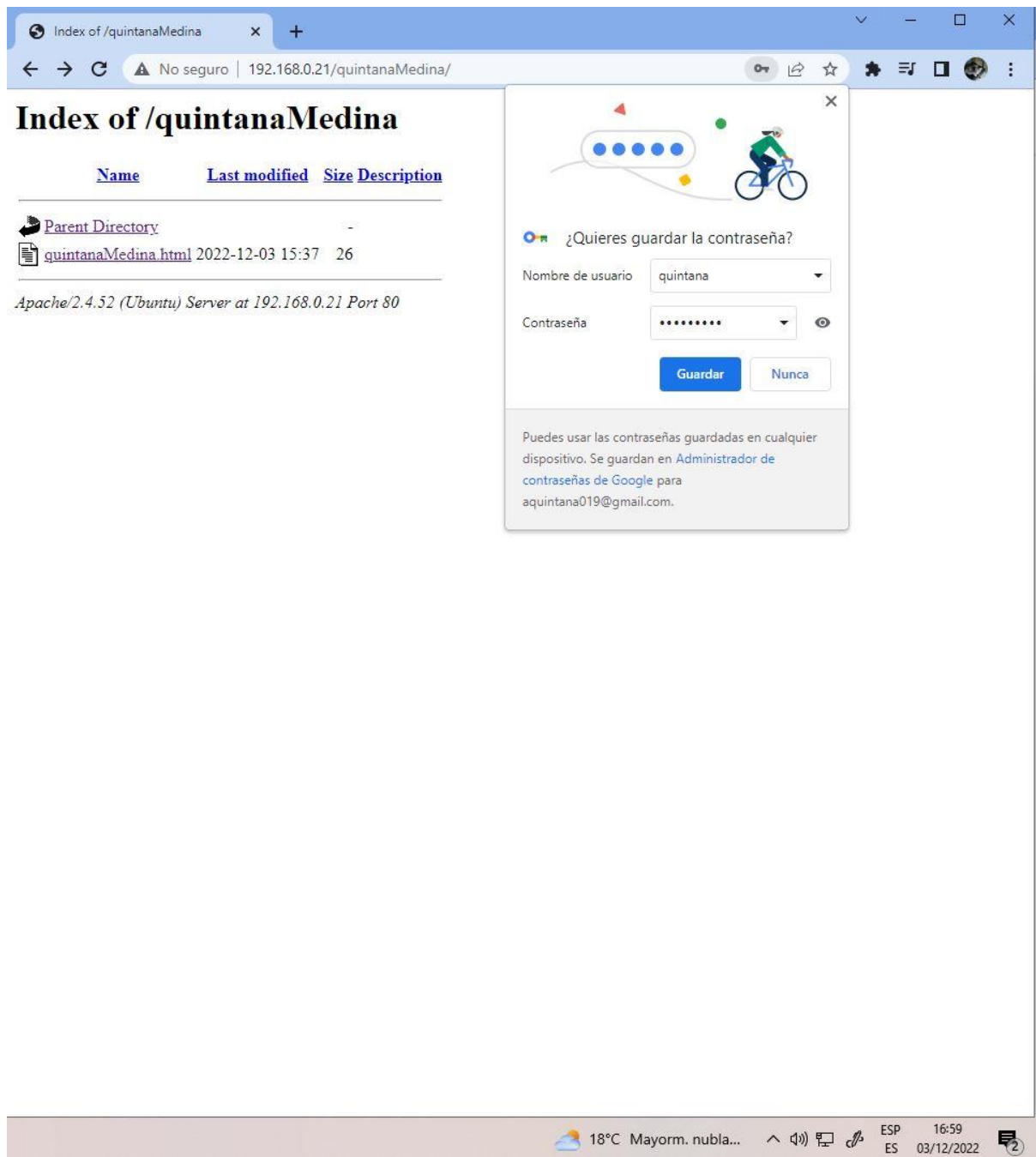
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

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

<Directory /var/www/html/quintanaMedina>
    DirectoryIndex quintana_medina.html
    Options Indexes FollowSymLinks
    AllowOverride None

    AuthType Basic
    AuthName "Acceso Restringido"
    AuthUserFile /etc/apache2/passwd
    Require user quintana medina

root@ububtuserveraqm:/var/www/html/quintanaMedina# sudo service apache2 restart
root@ububtuserveraqm:/var/www/html/quintanaMedina#
```

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

Mismo problema que en anteriores apartados, al hacerlo en casa no me era posible.

C.2) Autenticación Digest

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

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.

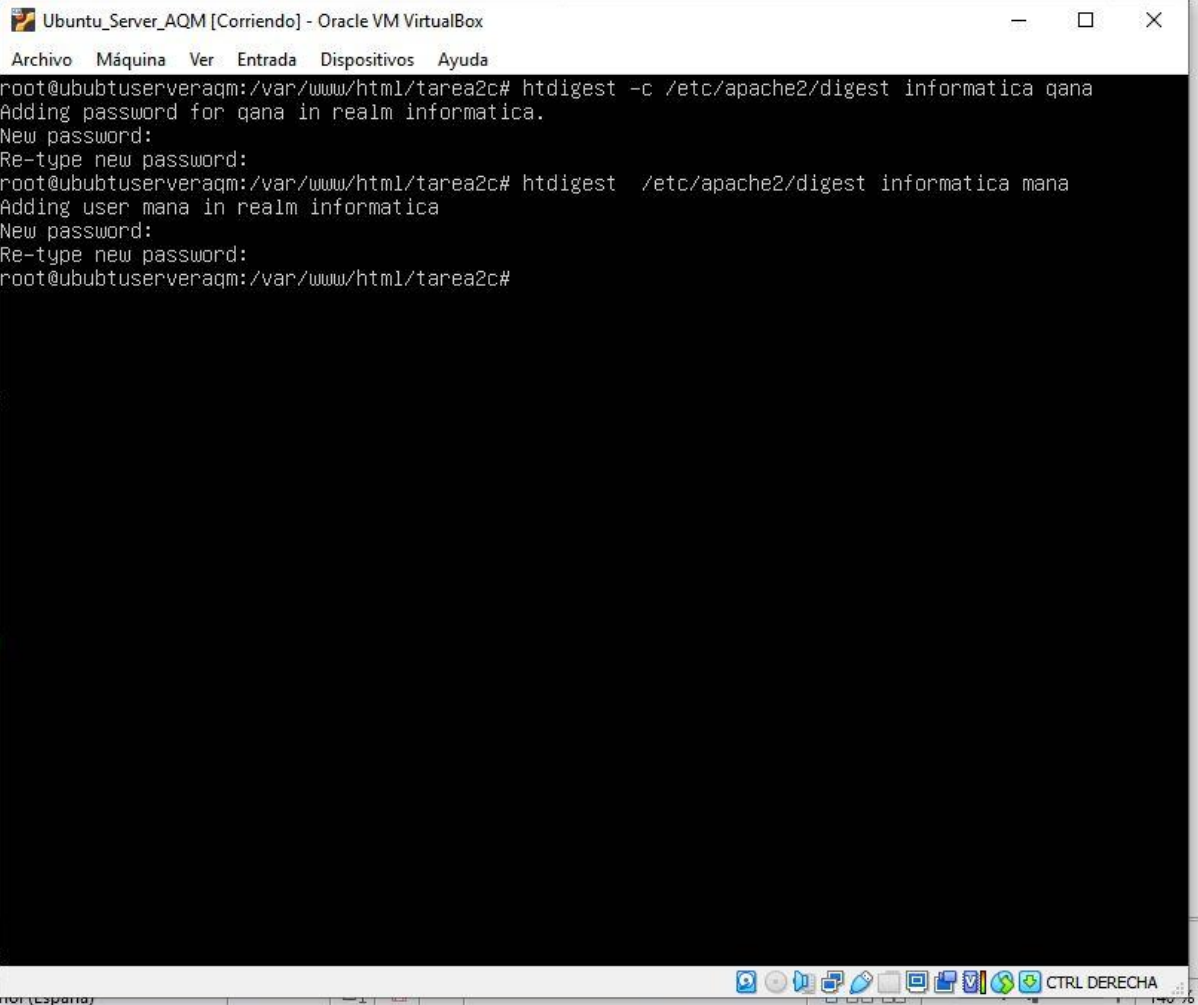
PASO 3) Para usar la autenticación Digest también hay que crear un fichero accesible (el fichero que se creará será también /etc/apache2/passwd pero para digest) en el que se guardarán los usuarios y contraseñas, pero esta vez asociados a un dominio (en el cuadro ejemplo de arriba el dominio o “realm”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

Ejemplo: Amapola Gutierrez de la Vega:

gamapola

vamapola

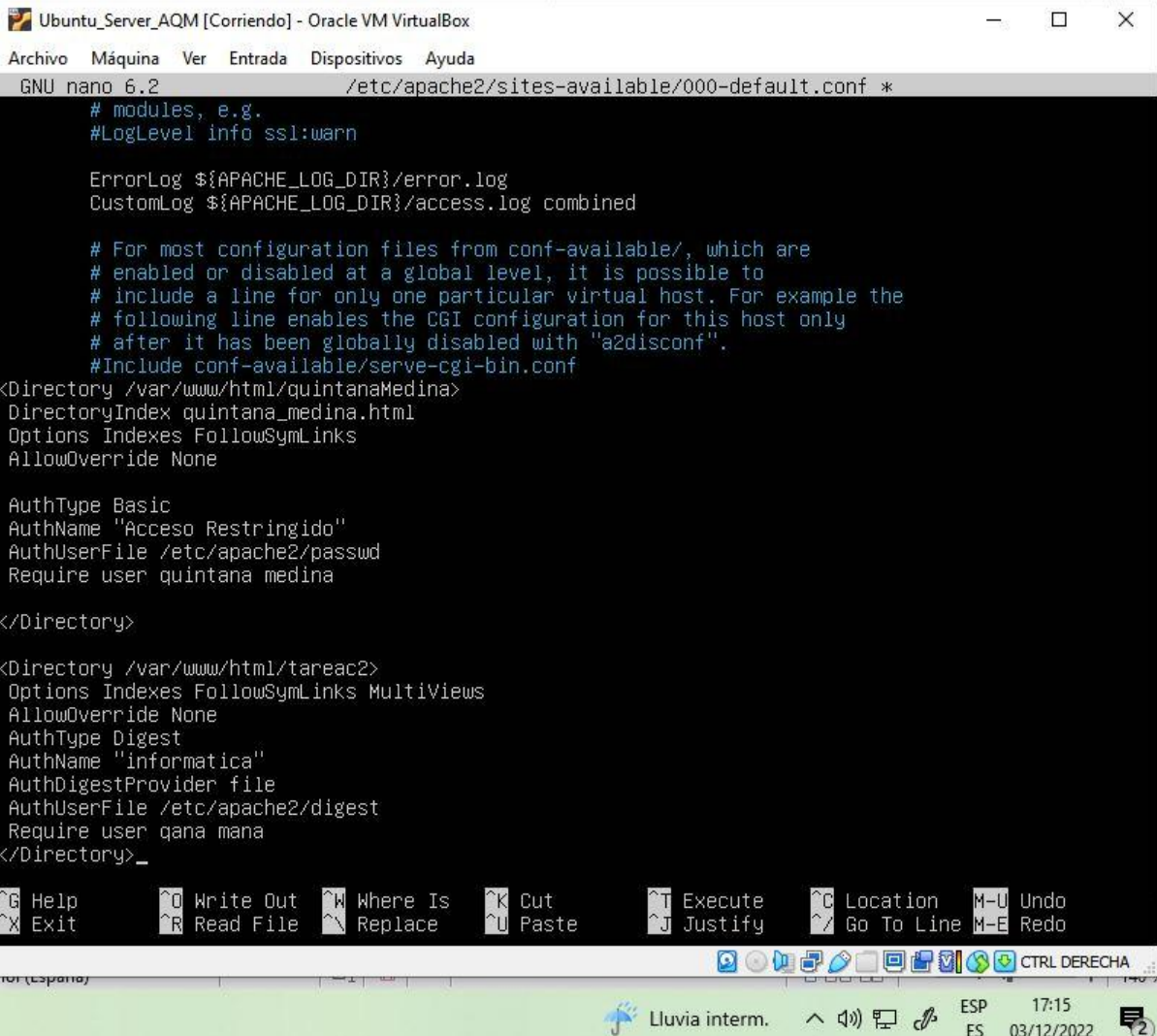
Siguiendo los mismos pasos que en el anterior ejemplo, pero esta vez usando htdigest como se indica, creamos los usuarios necesarios. Dejamos el realm como informática al igual que el ejemplo por comodidad.



```
Ubuntu_Server_AQM [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
root@ububtuserveraqm:/var/www/html/tarea2c# htdigest -c /etc/apache2/digest informatica qana
Adding password for qana in realm informatica.
New password:
Re-type new password:
root@ububtuserveraqm:/var/www/html/tarea2c# htdigest /etc/apache2/digest informatica mana
Adding user mana in realm informatica
New password:
Re-type new password:
root@ububtuserveraqm:/var/www/html/tarea2c#
```

The screenshot shows a terminal window titled "Ubuntu_Server_AQM [Corriendo] - Oracle VM VirtualBox". The terminal output shows the execution of the `htdigest` command to create two users, `qana` and `mana`, in the `informatica` realm. The prompt is `root@ububtuserveraqm:/var/www/html/tarea2c#`. The system prompts for a new password and its re-entry for each user. The desktop environment at the bottom shows a taskbar with various icons and a system tray indicating weather, volume, and date/time (17:09 on 03/12/2022).

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”.



```
GNU nano 6.2 /etc/apache2/sites-available/000-default.conf *
# modules, e.g.
#LogLevel info ssl:warn

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

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
<Directory /var/www/html/quintanaMedina>
  DirectoryIndex quintana_medina.html
  Options Indexes FollowSymLinks
  AllowOverride None

  AuthType Basic
  AuthName "Acceso Restringido"
  AuthUserFile /etc/apache2/passwd
  Require user quintana medina
</Directory>

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

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.

Pese a que he seguido los pasos indicados, no me ha salido el cuadro de autenticación por mucho que lo intentase, por lo que no puedo aportar captura de ello.

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

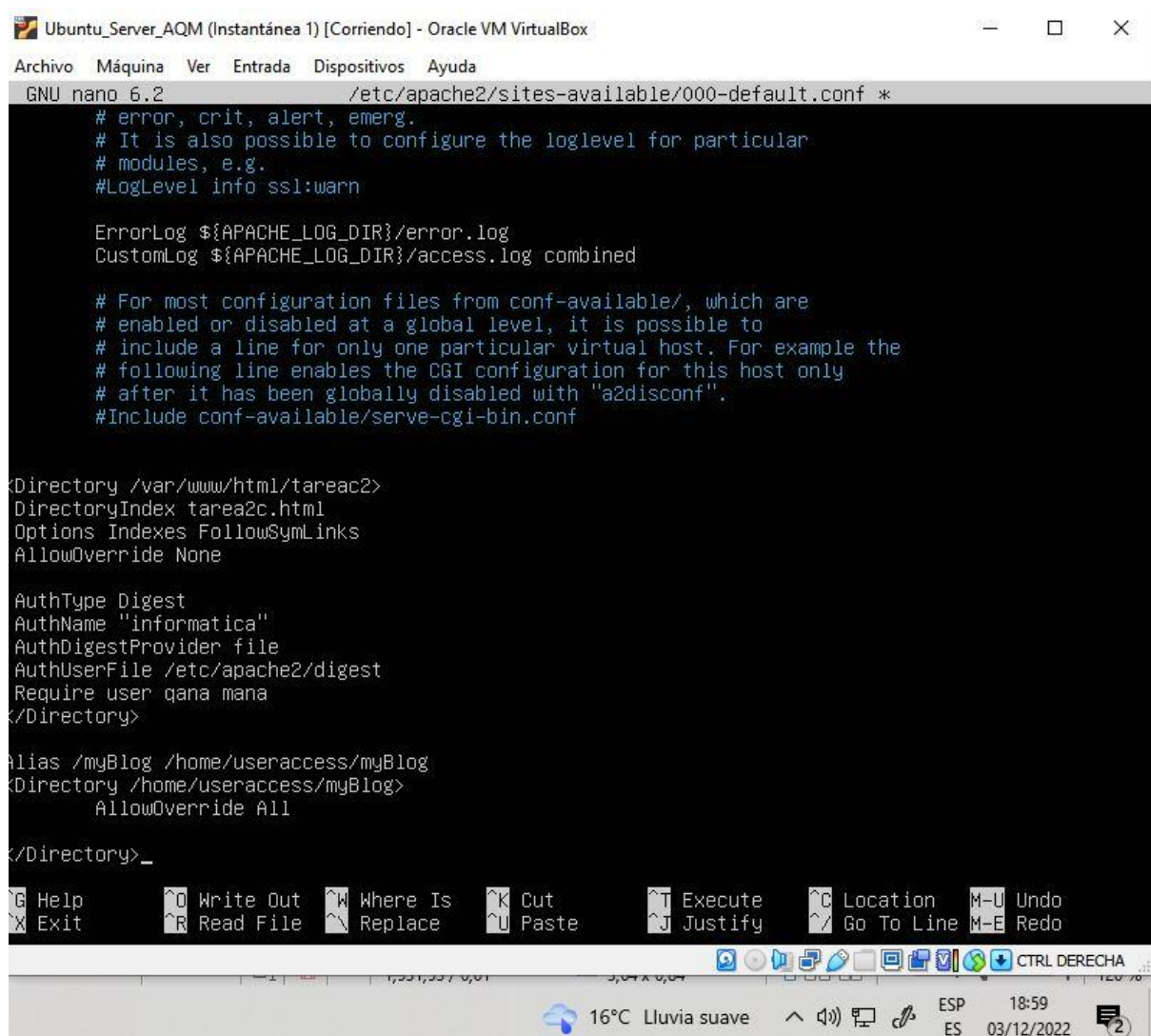
Toma una captura de los pasos 3, 4, 6 y 7 (de estas últimos una captura cuando sale el cuadro para autenticarte y luego una vez dentro del recurso /primo).

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

PASO 1) Crea el usuario 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.

Añadimos los elementos necesarios editando el archivo con nano.



```
GNU nano 6.2 /etc/apache2/sites-available/000-default.conf *
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

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

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

<Directory /var/www/html/tarea2c>
  DirectoryIndex tarea2c.html
  Options Indexes FollowSymLinks
  AllowOverride None

  AuthType Digest
  AuthName "informatica"
  AuthDigestProvider file
  AuthUserFile /etc/apache2/digest
  Require user gana mana
</Directory>

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

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 DERECHA
```

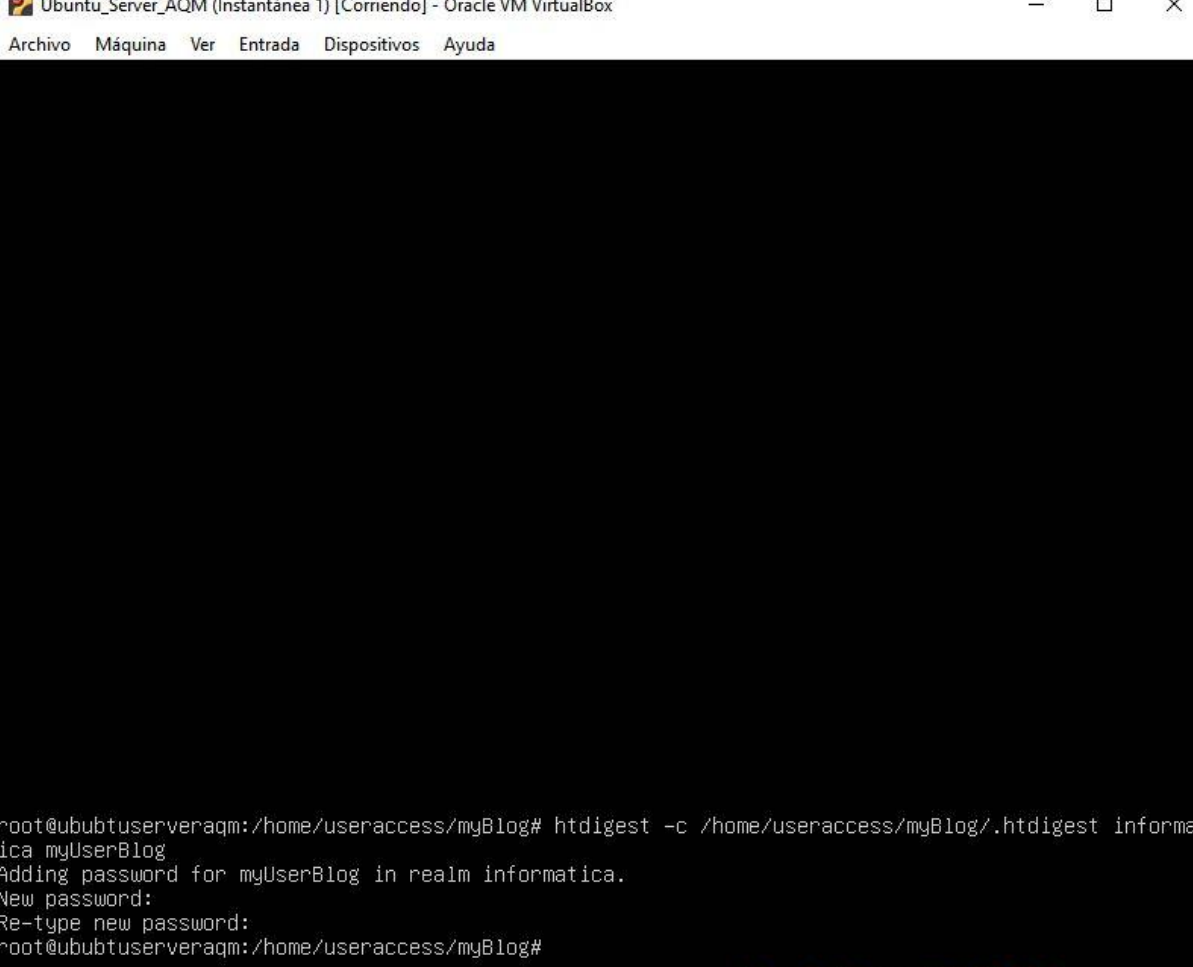
PASO 3) Reinicia el servidor para que los cambios tengan efecto.

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).

Al igual que hicimos antes, creamos en la dirección indicada el fichero .htdigest siguiendo las directivas necesarias:



The screenshot shows a terminal window titled "Ubuntu_Server_AQM (Instantánea 1) [Corriendo] - Oracle VM VirtualBox". The terminal output is as follows:

```
root@ububtuserveraqm:/home/useraccess/myBlog# htdigest -c /home/useraccess/myBlog/.htdigest informatica myUserBlog
Adding password for myUserBlog in realm informatica.
New password:
Re-type new password:
root@ububtuserveraqm:/home/useraccess/myBlog#
```

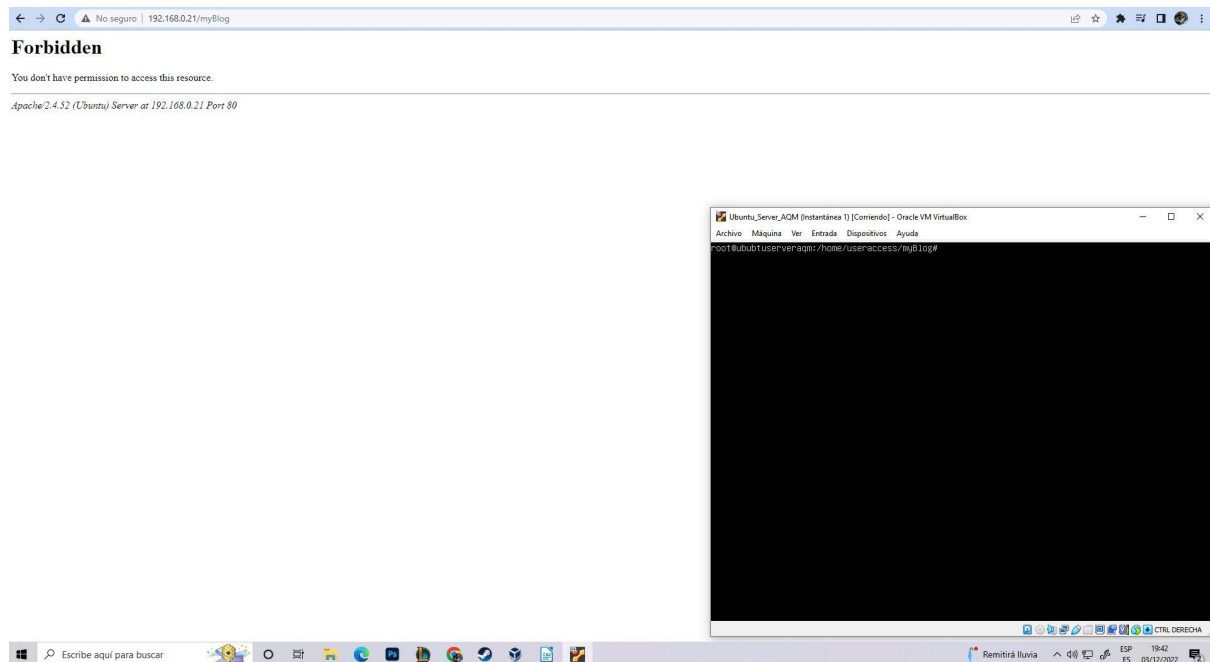
The terminal window has a menu bar with "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". The bottom of the window shows a taskbar with various icons and a system tray with the text "Remitirá lluvia", "ESP ES", "19:16", and "03/12/2022".

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).

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.

Lo hemos intentado, pero no permitía la entrada. Suponemos que lo habremos hecho mal en algún punto.



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

E) Ficheros de registros (logs)

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?

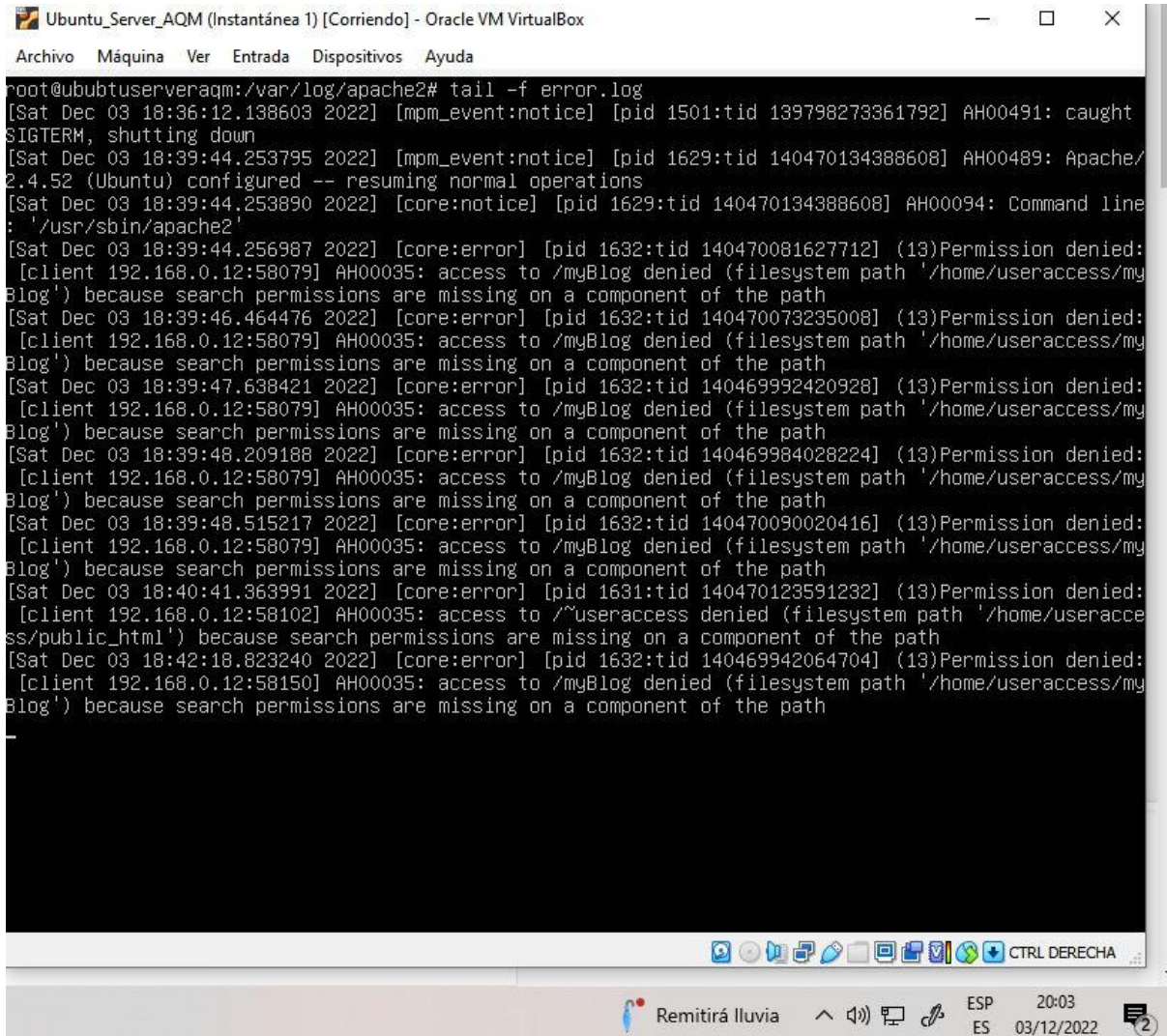
La directiva que marcar la ruta de archivo de errores no es otra que ErrorLog, que nos lleva, justamente, al archivo error.log, en el que se encuentran estos. Este tipo de elementos tienen una prioridad alta, dado que es el más importante de todos los registros, dado que se encarga de dar información sobre qué ha ido mal y la solución a dicho problema.

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

La ruta de archivo de los accesos, por su parte, viene dada por la directiva CustomLog, siendo su fichero access.log

PASO 2) Consulta el log de errores

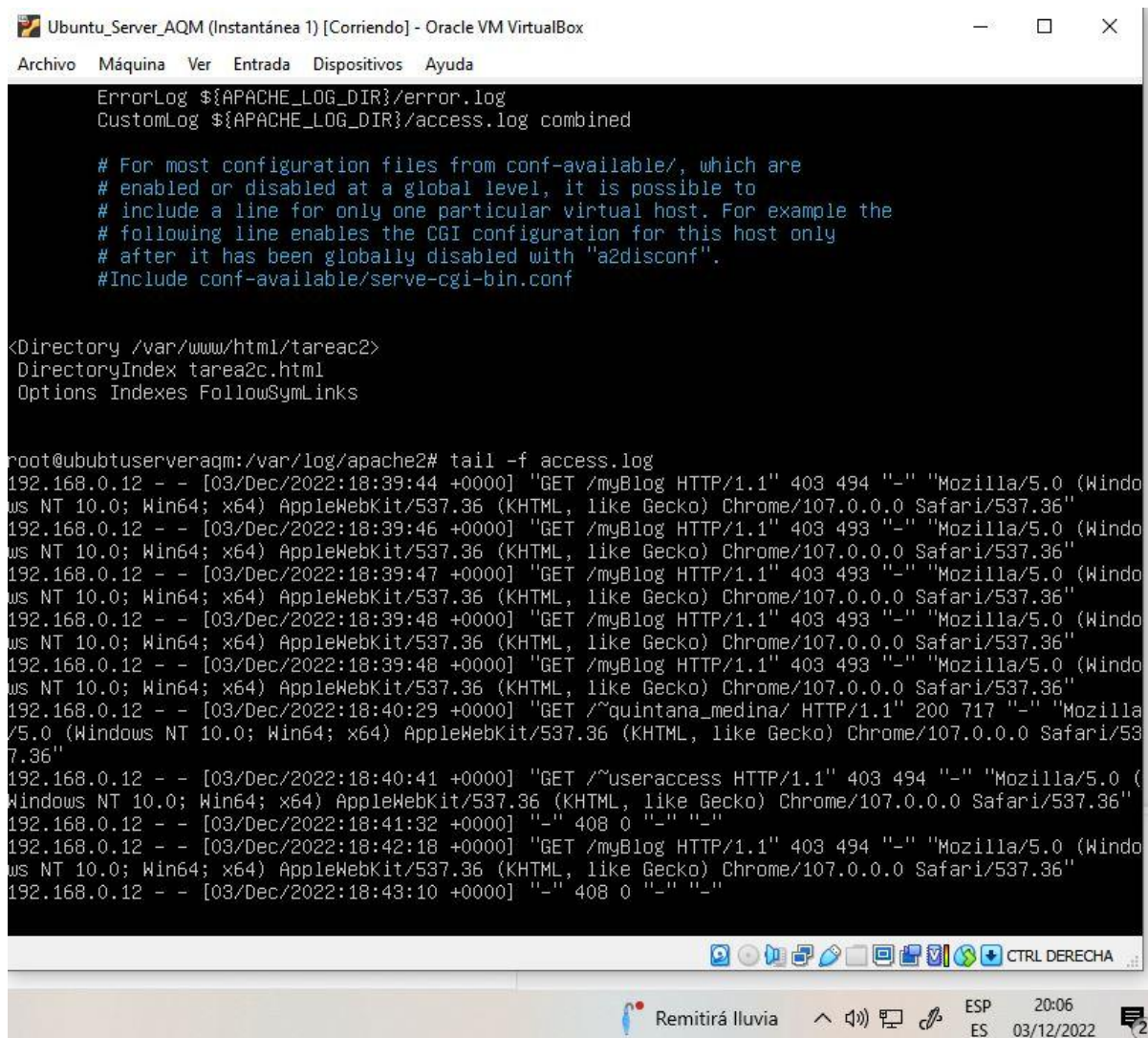
Para este paso, hemos tenido que encontrar la ruta del log de errores de Linux, en este caso situado en /var/log/apache2. Con eso en mente, nos ha bastado con usar el comando “tail -f” en el archivo error.log, teniendo el resultado deseado.



```
root@ububtuserveragm:/var/log/apache2# tail -f error.log
[Sat Dec 03 18:36:12.138603 2022] [mpm_event:notice] [pid 1501:tid 139798273361792] AH00491: caught
SIGTERM, shutting down
[Sat Dec 03 18:39:44.253795 2022] [mpm_event:notice] [pid 1629:tid 140470134388608] AH00489: Apache/
2.4.52 (Ubuntu) configured -- resuming normal operations
[Sat Dec 03 18:39:44.253890 2022] [core:notice] [pid 1629:tid 140470134388608] AH00094: Command line
: '/usr/sbin/apache2'
[Sat Dec 03 18:39:44.256987 2022] [core:error] [pid 1632:tid 140470081627712] (13)Permission denied:
[client 192.168.0.12:58079] AH00035: access to /myBlog denied (filesystem path '/home/useraccess/my
Blog') because search permissions are missing on a component of the path
[Sat Dec 03 18:39:46.464476 2022] [core:error] [pid 1632:tid 140470073235008] (13)Permission denied:
[client 192.168.0.12:58079] AH00035: access to /myBlog denied (filesystem path '/home/useraccess/my
Blog') because search permissions are missing on a component of the path
[Sat Dec 03 18:39:47.638421 2022] [core:error] [pid 1632:tid 140469992420928] (13)Permission denied:
[client 192.168.0.12:58079] AH00035: access to /myBlog denied (filesystem path '/home/useraccess/my
Blog') because search permissions are missing on a component of the path
[Sat Dec 03 18:39:48.209188 2022] [core:error] [pid 1632:tid 140469984028224] (13)Permission denied:
[client 192.168.0.12:58079] AH00035: access to /myBlog denied (filesystem path '/home/useraccess/my
Blog') because search permissions are missing on a component of the path
[Sat Dec 03 18:39:48.515217 2022] [core:error] [pid 1632:tid 140470090020416] (13)Permission denied:
[client 192.168.0.12:58079] AH00035: access to /myBlog denied (filesystem path '/home/useraccess/my
Blog') because search permissions are missing on a component of the path
[Sat Dec 03 18:40:41.363991 2022] [core:error] [pid 1631:tid 140470123591232] (13)Permission denied:
[client 192.168.0.12:58102] AH00035: access to /~useraccess denied (filesystem path '/home/useracce
ss/public_html') because search permissions are missing on a component of the path
[Sat Dec 03 18:42:18.823240 2022] [core:error] [pid 1632:tid 140469942064704] (13)Permission denied:
[client 192.168.0.12:58150] AH00035: access to /myBlog denied (filesystem path '/home/useraccess/my
Blog') because search permissions are missing on a component of the path
-
```

PASO 3) Consulta el log de accesos

Por su parte, para el log de accesos no tenemos que irnos muy lejos, ya que el archivo access.log (donde se encuentra el archivo de accesos) está en esta localización. Aquí, usando el mismo comando que en el paso 2, podremos ver lo requerido.



The screenshot shows a terminal window titled "Ubuntu_Server_AQM (Instantánea 1) [Corriendo] - Oracle VM VirtualBox". The terminal displays the configuration of the Apache web server, including the location of the error and access logs, and the configuration for the directory `/var/www/html/tarea2c`. Below the configuration, the command `tail -f access.log` is executed, showing a series of HTTP GET requests from various user agents, including Mozilla/5.0 and Chrome/107.0.0.0.

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

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

<Directory /var/www/html/tarea2c>
    DirectoryIndex tarea2c.html
    Options Indexes FollowSymLinks

```

```
root@ububtuserveraqm:/var/log/apache2# tail -f access.log
192.168.0.12 - - [03/Dec/2022:18:39:44 +0000] "GET /myBlog HTTP/1.1" 403 494 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36"
192.168.0.12 - - [03/Dec/2022:18:39:46 +0000] "GET /myBlog HTTP/1.1" 403 493 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36"
192.168.0.12 - - [03/Dec/2022:18:39:47 +0000] "GET /myBlog HTTP/1.1" 403 493 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36"
192.168.0.12 - - [03/Dec/2022:18:39:48 +0000] "GET /myBlog HTTP/1.1" 403 493 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36"
192.168.0.12 - - [03/Dec/2022:18:39:48 +0000] "GET /myBlog HTTP/1.1" 403 493 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36"
192.168.0.12 - - [03/Dec/2022:18:40:29 +0000] "GET /~quintana_medina/ HTTP/1.1" 200 717 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36"
192.168.0.12 - - [03/Dec/2022:18:40:41 +0000] "GET /~useraccess HTTP/1.1" 403 494 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36"
192.168.0.12 - - [03/Dec/2022:18:41:32 +0000] "-" 408 0 "-" "-"
192.168.0.12 - - [03/Dec/2022:18:42:18 +0000] "GET /myBlog HTTP/1.1" 403 494 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36"
192.168.0.12 - - [03/Dec/2022:18:43:10 +0000] "-" 408 0 "-" "-"
```

F) Módulos status e info

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

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.

Hemos sintetizado las capturas de los pasos 2 y 4 en una sola. Para lograr esto, hemos entrado en `/etc/apache2/mods-available/status.conf`, añadiendo la ip de la máquina

anfitriona para poder ver el estado.

[illegible]

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

Nuevamente, sintetizamos las dos capturas requeridas. Mismo procedimiento que el anterior ejercicio, solo que entrando en info.conf.

[←](#)
[→](#)
[No logro | 192.168.0.21/server-info](#)

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_authn_core.c, mod_authn_host.c, mod_authn_user.c, mod_autoindex.c, mod_deflate.c, mod_dir.c, mod_env.c, mod_filter.c, mod_inflate.c, mod_log_config.c, mod_logio.c, mod_mime.c, mod_negotiation.c, mod_remoteip.c, mod_rewrite.c, mod_socshandlers.c, mod_status.c, mod_unixd.c, mod_userdir.c, mod_version.c, mod_watchdog.c.
```

Server Settings

Server Version: Apache/2.4.52 (Ubuntu)
Server Built: 2022-09-30T04:09:50
Server loaded APR Version: 1.7.0
Compiled with APR Version: 1.7.0
Server loaded APU Version: 1.6.1
Compiled with APU Version: 1.6.1
Module Magic Number: 20120211121
Hostname/port: 192.168.0.21:80
Timeouts: connection: 300 keep-alive: 5
MPM Name: event
MPM Information: Max Daemons: 2 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
- D APR_HAS_MMAP
- D APR_HAVE_IPV6 (IPv4-mapped addresses enabled)
- D APR_USE_PTHREAD_SEMANTICS
- D SINGLE_LISTEN_UNSERIALIZED_ACCEPT
- D APR_HAS_OTHER_CHILD
- D AP_HAVE_RELIABLE_PIPED_LOGS
- D HTTPD_ROOT="/etc/apache2"
- D SUEXEC_BIN="/usr/lib/apr/apr2/apache2/suexec"
- D DEFAULT_PIDLOG="/var/run/apache2.pid"
- D DEFAULT_SCOREBOARD="logs/apache_runtime_status"
- D DEFAULT_ERRORLOG="logs/error_log"
- D AP_TYPES_CONFIG_FILE="mime.types"
- D SERVER_CONFIG_FILE="apache2.conf"

```

Ubuntu Server 22.04 LTS [Command]: Oracle VM VirtualBox
Archive Manager
Vista
Disks
Devices
Apptools
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 192.0.2.0/24
    Require ip 192.168.0.21/24
</Location>

</IfModule>

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

root@ubuntu-server-apache:/etc/apache2# sudo service apache2 restart
root@ubuntu-server-apache:/etc/apache2#
        
```

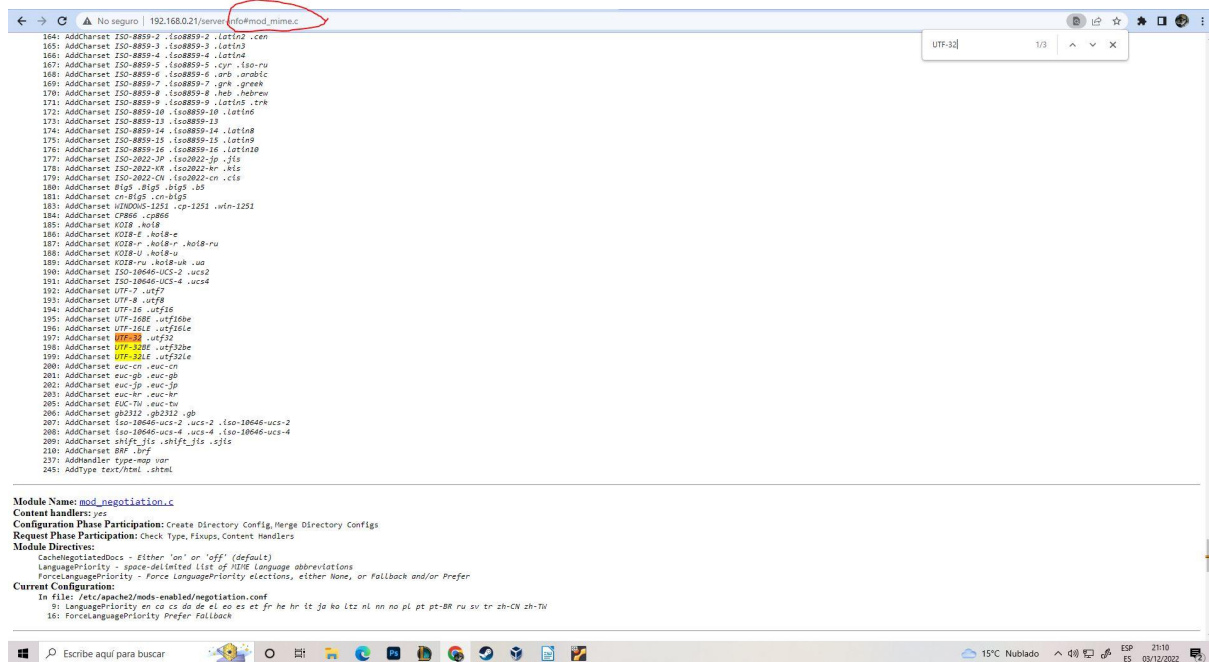
Startup Hooks

Pre-Config:

```
#!/bin/sh
# ...
```

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?

Tal y como podemos comprobar en la siguiente captura, sí, tenemos dicho módulo cargado junto a esa configuración de caracteres.



```
164: AddCharset ISO-8859-2 .iso8859-2 .latin2 .csm
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 .grec .greek
170: AddCharset ISO-8859-8 .iso8859-8 .heb .hebrew
171: AddCharset ISO-8859-9 .iso8859-9 .latin .tck
172: AddCharset ISO-8859-10 .iso8859-10 .latin6
173: AddCharset ISO-8859-11 .iso8859-11
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 .cls
180: AddCharset Big5 .big5 .big5 .os
181: AddCharset cn-big5 .cn-big5
182: AddCharset UTF-8 .utf8 .utf8
183: AddCharset UTF-16 .utf16 .utf16
184: AddCharset UTF-32 .utf32 .utf32
185: AddCharset KOI8-R .koi8-r
186: AddCharset KOI8-U .koi8-u
187: AddCharset KOI8-RU .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-16LE .utf16le
196: AddCharset UTF-32 .utf32
197: AddCharset UTF-32LE .utf32le
198: AddCharset UTF-32 .utf32
199: AddCharset UTF-32LE .utf32le
200: AddCharset euc-cn .euc-cn
201: AddCharset euc-gb .euc-gb
202: AddCharset euc-jp .euc-jp
203: AddCharset euc-kr .euc-kr
204: AddCharset EUC-TW .euc-tw
205: AddCharset gb2312 .gb2312 .gb
206: AddCharset iso-10646-ucs-2 .ucs-2 .iso-10646-ucs-2
207: AddCharset iso-10646-ucs-4 .ucs-4 .iso-10646-ucs-4
208: AddCharset shift_jis .shift_jis .sjis
209: AddCharset BDF .bdf
210: AddHandler type-map var
211: AddType text/html .html

Module Name: mod_negotiation.c
Content handlers: yes
Configuration Phase Participation: Create Directory Config, Merge Directory Configs
Request Phase Participation: Check Type, Fixups, Content Handlers
Module Directives:
CacheNegotiatedDocs: Either 'on' or 'off' (default)
LanguagePriority: space-separated list of 128 language abbreviations
ForceLanguagePriority: Force LanguagePriority elections, either None, or fallback and/or Prefer
Current Configuration:
In File: /etc/apache2/mods-enabled/negotiation.conf
9: LanguagePriority en cs da de el eo es et fr he hr it ja ko ltz nl nn no pl pt pt-BR ru sv tr zh-CN zh-TW
16: ForceLanguagePriority Prefer Fallback
```

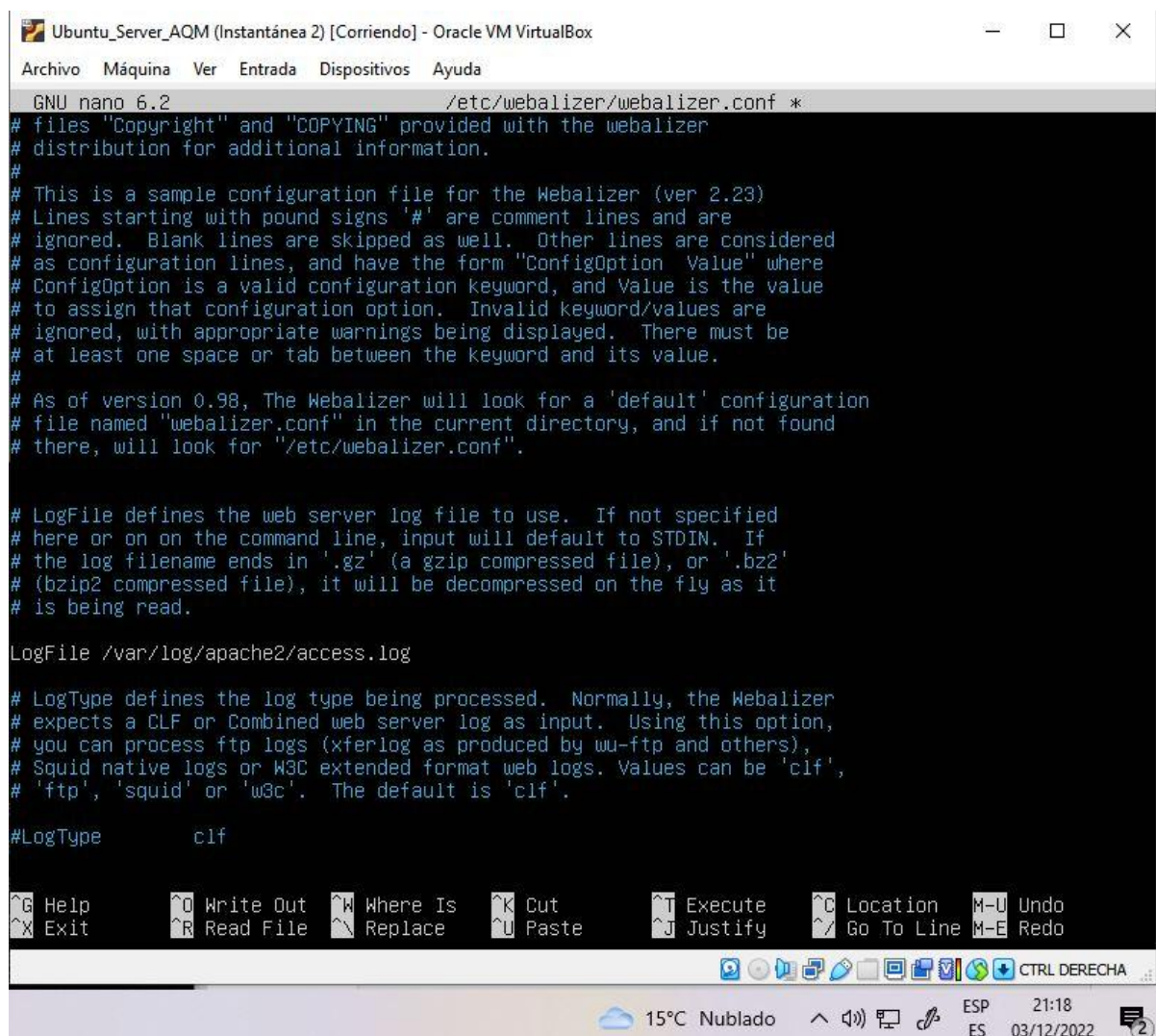
Toma una captura de los pasos 6 y 8.

G) Webalizer

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.

Como era de esperar, la ruta no es la correcta. La modificamos poniendo en su lugar la misma que usamos algunos ejercicios más arriba.



```
GNU nano 6.2 /etc/webalizer/webalizer.conf *
# 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
#
# 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
```

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.

La ubicación no era la correcta, así que haciendo uso de las indicaciones de más abajo hemos podido solucionarlo.

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/
```

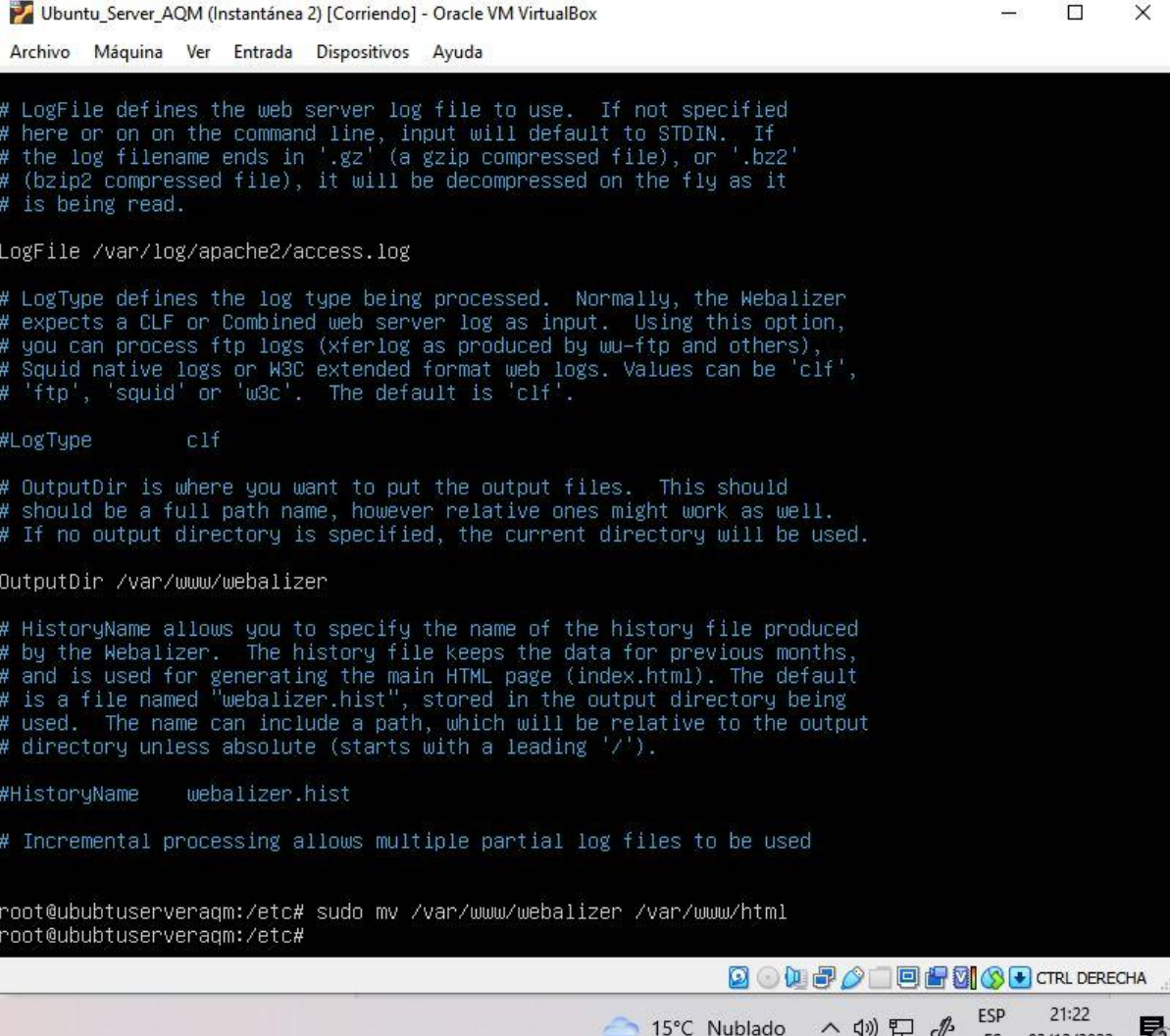
A continuación, vamos a editar el archivo de configuración de Webalizer introduce la siguiente instrucción:

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

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.



```
Ubuntu_Server_AQM (Instantánea 2) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

# 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

# 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/webalizer

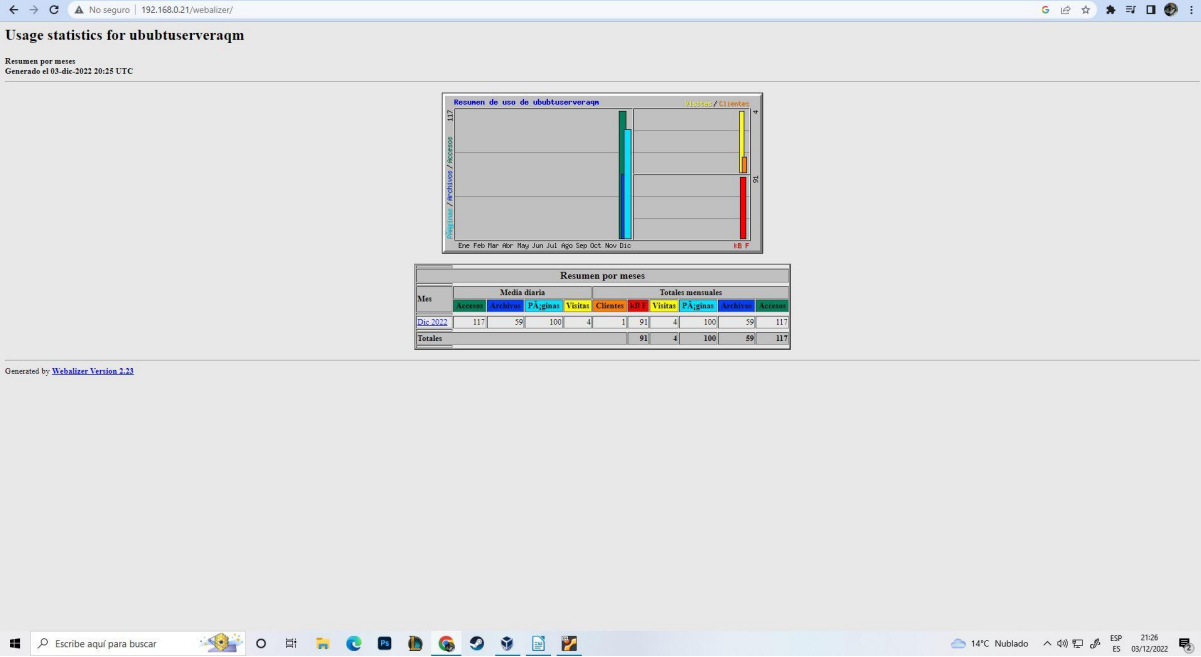
# HistoryName allows you to specify the name of the history file produced
# by the Webalizer.  The history file keeps the data for previous months,
# and is used for generating the main HTML page (index.html).  The default
# is a file named "webalizer.hist", stored in the output directory being
# used.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

#HistoryName   webalizer.hist

# Incremental processing allows multiple partial log files to be used

root@ububtuserveraqm:/etc# sudo mv /var/www/webalizer /var/www/html
root@ububtuserveraqm:/etc#
```

15°C Nublado 21:22 ESP 03/12/2022



Toma una captura de los pasos 2 y 5.

F) GitHub