Hoja02_Apache_04

1.- Vamos a utilizar el control de acceso básico que tiene Apache.

a) Para ello vamos a permitir <u>únicamente</u> que acceda a nuestro servidor la propia máquina física. Prueba a acceder desde la máquina física para confirmar que la configuración es correcta. Comprueba también que no puedes acceder desde el navegador de la propia máquina virtual.

b) Ahora implementa una regla que no permita el acceso a la web en caso de que la petición provenga del equipo real. Comprueba el resultado con el navegador desde la máquina virtual y desde la máquina real.

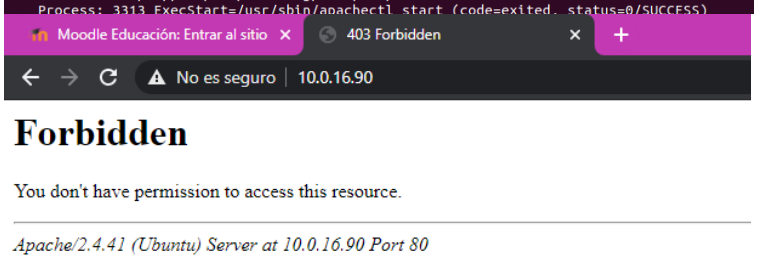
```
        AllowOverride None
        Require all granted
</Directory>

<Directory /var/www/>
        Options Indexes FollowSymLinks
        AllowOverride None
        #Require all granted
        #Ejercicio 1
        #Required ip 10.0.16.38

        #Ejercicio 2
        <RequireAll>
                Require all granted
                Required not ip 10.0.16.38
        </RequireAll>
</Directory>

#<Directory /srv/>
#       Options Indexes FollowSymLinks
```

```
[1]+  Detenido                 sudo nano apache2.conf
usuario@usuario-daw2:/etc/apache2$ sudo nano apache2.conf
usuario@usuario-daw2:/etc/apache2$ sudo systemctl stop apache2
usuario@usuario-daw2:/etc/apache2$ sudo systemctl start apache2
Job for apache2.service failed because the control process exited with error code.
See "systemctl status apache2.service" and "journalctl -xe" for details.
usuario@usuario-daw2:/etc/apache2$ sudo nano apache2.conf
usuario@usuario-daw2:/etc/apache2$ sudo systemctl stop apache2
usuario@usuario-daw2:/etc/apache2$ sudo systemctl start apache2
usuario@usuario-daw2:/etc/apache2$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2020-11-13 19:25:23 CET; 12s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 3313 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
```



Forbidden

You don't have permission to access this resource.

Apache/2.4.41 (Ubuntu) Server at 10.0.16.90 Port 80

c) Dentro de la misma sección, filtra el acceso por hora permitiendo solo desde las 14 hasta las 20. Comprueba que puedes acceder. Ahora cámbialo para estar fuera del rango adecuado en el momento en el que realiza la práctica y comprueba que no puedes acceder.

```
<Directory /var/www/>
        Options Indexes FollowSymLinks
        AllowOverride None
        #Require all granted
        #Ejercicio 1
        #Required ip 10.0.16.38

        #Ejercicio 2
        <RequireAll>
                Require all granted
                Require not ip 10.0.16.38
        </RequireAll>
        #Ejercicio c
        Require expr %{TIME_HOUR} -gt 10 && %{TIME_HOUR} -lt 14

        #Ejercicio d
        Require expr "%{HTTP_USER_AGENT}=~/Chrome/"

^G Ver ayuda ^O Guardar   ^W Buscar    ^K Cortar Tex^J Justificar^C Posición
^X Salir     ^R Leer fich.^\ Reemplazar^U Pegar     ^T Ortografía^_ Ir a línea
```

d) Dentro de la misma sección. Permite solo las conexiones desde un navegador. Para ello debes utilizar la directiva:

Require expr %{HTTP_USER_AGENT}=='STRING DE NAVEGADOR'

Busca en Internet cuál sería el HTTP_USER_AGENT para distintos navegadores

e) Trata de realizar las siguientes combinaciones utilizando <RequireAll>, <RequireAny> y <RequireNone>:

a. Que solo se pueda acceder desde Chrome Y a unas horas determinadas.
b. Que solo se pueda acceder desde la IP de la máquina real O desde Chrome.
c. Que solo se pueda acceder desde Chrome O a una hora determinada O desde la máquina real

2.- Ahora vamos a utilizar la autenticación que nos ofrece Apache.

a) Añade el módulo de autenticación **básica**. Si ya existe, comprueba que se encuentra en el directorio de módulos activos.

En el dominio despliegue.com crea un directorio llamado **privado**.

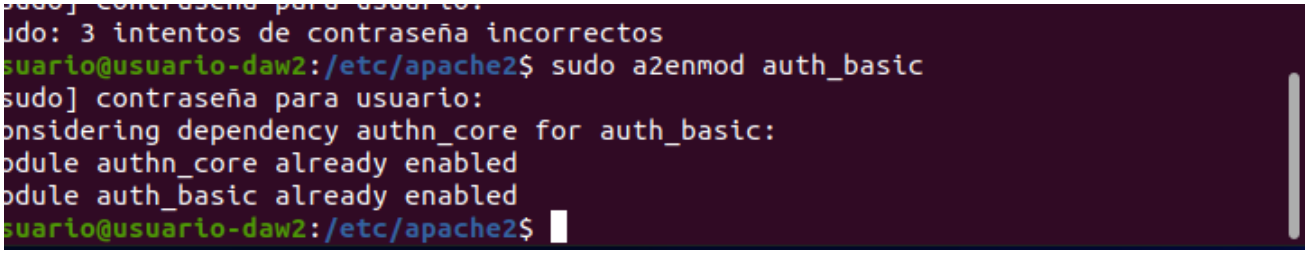
```
-rw-r--r-- 1 root root   74 abr 13  2020 vhost_alias.load
-rw-r--r-- 1 root root   66 abr 13  2020 xml2enc.load
usuario@usuario-daw2:/etc/apache2/mods-available$ cd /var/www/despliegue/
usuario@usuario-daw2:/var/www/despliegue$ sudo mkdir privado
[sudo] contraseña para usuario:
usuario@usuario-daw2:/var/www/despliegue$ cd privado/
usuario@usuario-daw2:/var/www/despliegue/privado$
```

b) Crea un fichero de usuarios y crea a homer y bart.

```
-rw-r--r-- 1 root root   17 nov  2 19:22 despliegueJoseba.html
drwxr-xr-x 2 root root 4096 nov 13 20:27 privado
usuario@usuario-daw2:/var/www/despliegue$ cd privado
usuario@usuario-daw2:/var/www/despliegue/privado$ cd /etc/apache2/
usuario@usuario-daw2:/etc/apache2$ sudo htpasswd -c /etc/apache2/passwd homer
[sudo] contraseña para usuario:
New password:
Re-type new password:
Adding password for user homer
usuario@usuario-daw2:/etc/apache2$ sudo htpasswd -c /etc/apache2/passwd bart
New password:
Re-type new password:
Adding password for user bart
usuario@usuario-daw2:/etc/apache2$
```

c) Permite el acceso al directorio privado que has creado a los usuarios homer y bart.

d) Comprueba que el acceso se realiza de manera autenticada.



3.- Prueba ahora a utilizar la autenticación **digest** y crea otro un directorio **privado** pero esta vez en dwes.com en el que uses este tipo de autenticación para poder acceder a él.
En este caso hay que crear un archivo de credenciales Digest. Para ello se debe utilizar el comando **htdigest**.

4.- Por último, abre los dos archivos de contraseñas con un editor de texto plano y comprueba que metiendo la misma contraseña no se codifica igual con Basic que con Digest.