

Practica 4



ADMINISTRACIÓN DE APACHE II

Paulo Gustavo Soares Teixeira

A) Control de acceso por IP y nombre de dominio.

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

PASO 1) Comprueba si está habilitado el módulo authz_host. ¿Lo está? NO

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 el servidor Windows 2008.

```
<Directory /var/www/html/paulo>
    DirectoryIndex paulo.html
    Options FollowSymLinks
    AllowOverride None
    Require ip 172.26.2.121
</Directory>
```

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

```
soares_teixeira@servidorLinux21:/var/www/html/paulo$ sudo service apache2 restart
* Restarting web server apache2
soares_teixeira@servidorLinux21:/var/www/html/paulo$
```

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



PASO 6) Abre un navegador desde tu máquina servidor Windows 2008 e intenta acceder al recurso /tuNombre/ y comprueba que no se puede.

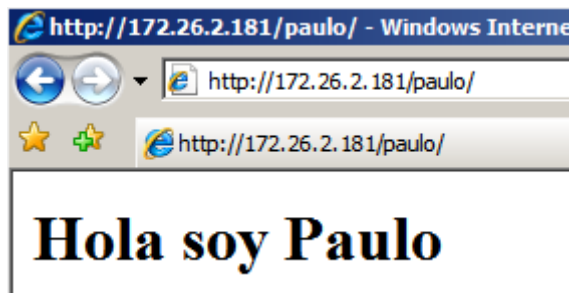


PASO 7) Añade el acceso al recurso de tu carpeta para el servidor windows 2008 pero usando su nombre de host en vez de su IP.

```
<Directory /var/www/html/paulo>
    DirectoryIndex paulo.html
    Options FollowSymLinks
    AllowOverride None
    Require ip 172.26.2.121
    Require host SERVIDORW821.daw21.com
</Directory>
```

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

PASO 9) Abre un navegador desde tu máquina servidor Windows 2008 e intenta acceder al recurso /tuNombre/ y comprueba que ahora sí se puede.



B) Autenticación y autorización Basic y Digest.

B1) Autenticación Basic:

Toma capturas de los pasos 3,4, 6 y 7.

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

PASO 2) Vamos a crear el directorio /amigo/ dentro de nuestro directorio raíz /var/www/html/. Dentro añadiremos un archivo amigo.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 amigo1 y amigo2.

```
soares_teixeira@servidorLinux21:/var/www/html/paulo$ sudo htpasswd -c /etc/apache2/passwd amigo1
New password:
Re-type new password:
Adding password for user amigo1
soares_teixeira@servidorLinux21:/var/www/html/paulo$ sudo htpasswd /etc/apache2/passwd amigo2
New password:
Re-type new password:
Adding password for user amigo2
soares_teixeira@servidorLinux21:/var/www/html/paulo$
```

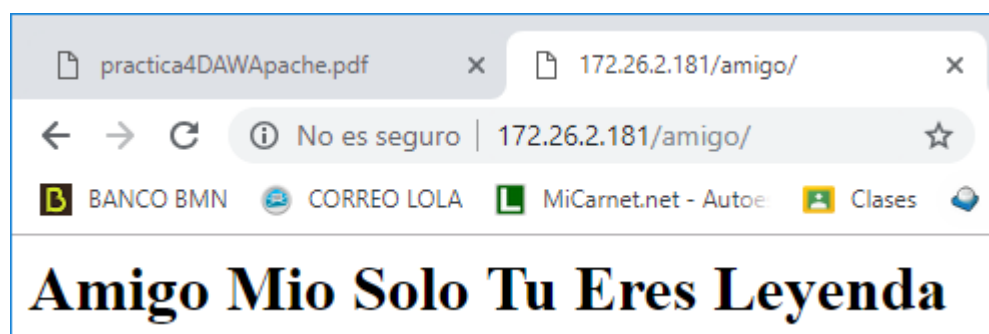
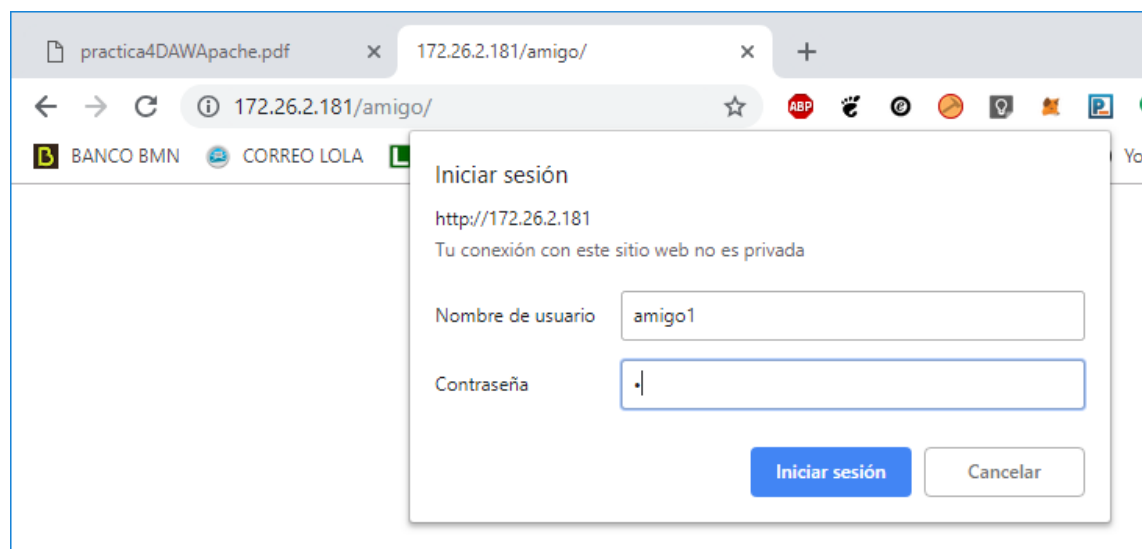
PASO 4) Edita el fichero de configuración `/etc/apache2/sites-available/000-default.conf` y permite el acceso al directorio `/var/www/html/amigo` a los usuarios `amigo1` y `amigo2` (ver cuadro ejemplo arriba).

```
<Directory /var/www/html/amigo>
    DirectoryIndex amigo.html
    Options Indexes FollowSymLinks Multiviews
    AllowOverride None
    Order allow,deny

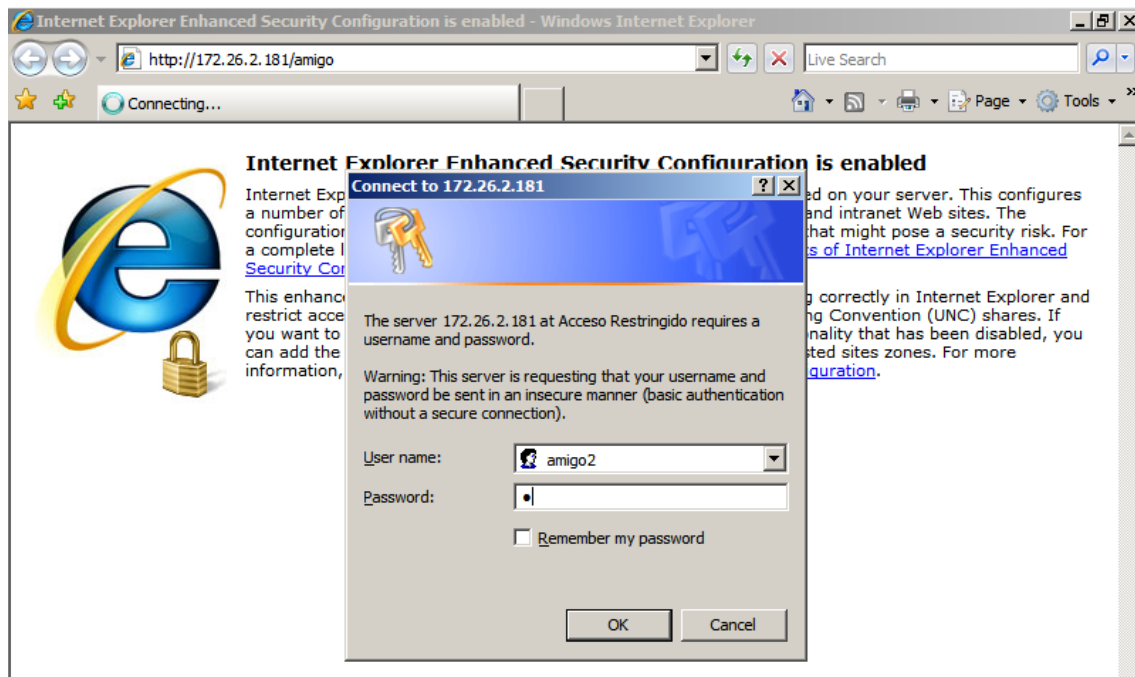
    allow from 172.26.2.121
    allow from SERVIDORW821.daw21.com
    AuthType Basic
    AuthName "Acceso Restringido"
    AuthUserFile /etc/apache2/passwd
    Require user amigo1 amigo2
</Directory>
```

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

PASO 6) Abre un navegador desde tu máquina física y accede al recurso `/amigo` como usuario `amigo1`.



PASO 7) Abre un navegador desde tu servidor Windows 2008 y accede al recurso /amigo como usuario amigo2.



B2) Autenticación Digest:

Toma capturas de los pasos 3,4, 6 y 7.

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

PASO 2) Vamos a crear el directorio /primo/ dentro de nuestro directorio raíz /var/www/html/. Dentro añadiremos un archivo primo.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 primo1 y primo2.

```
soares_teixeira@servidorLinux21:/var/www/html$ sudo htdigest -c /etc/apache2/digest informatica primo1
Adding password for primo1 in realm informatica.
New password:
Re-type new password:
soares_teixeira@servidorLinux21:/var/www/html$ sudo htdigest /etc/apache2/digest informatica primo2
Changing password for user primo2 in realm informatica
New password:
Re-type new password:
soares_teixeira@servidorLinux21:/var/www/html$ _
```

PASO 4) Edita el fichero de configuración /etc/apache2/sites-available/000-default.conf y permite el acceso al directorio /var/www/html/primo a los usuarios primo1 y primo2 (ver cuadro ejemplo arriba).

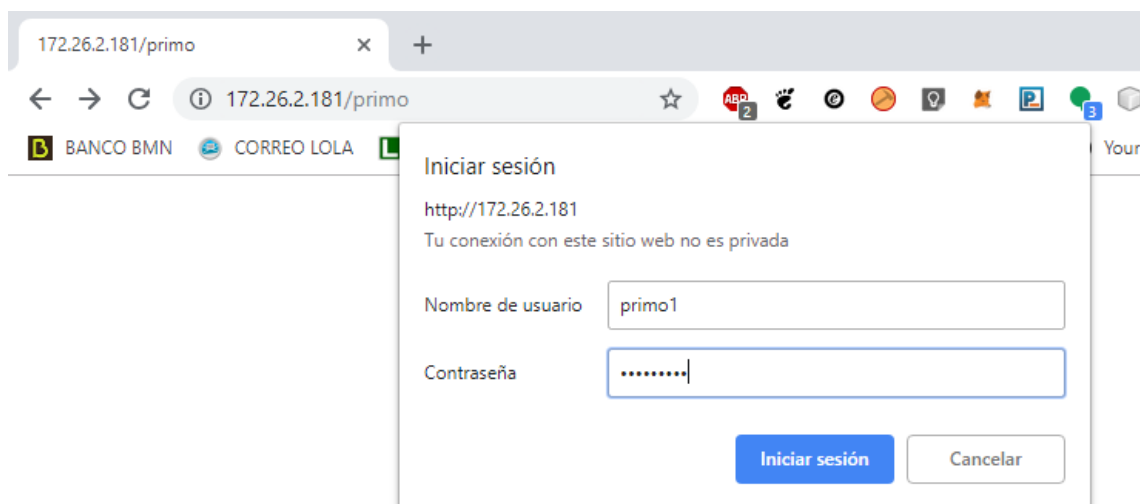
```
<Directory /var/www/html/primo>
    DirectoryIndex primo.html
    Options Indexes FollowSymLinks Multiviews
    AllowOverride None
    Order allow,deny

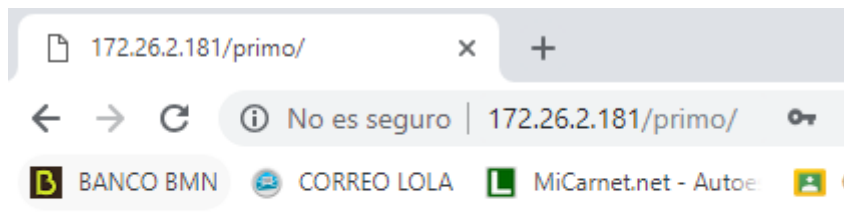
    allow from 172.26.2.121
    allow from SERVIDORW821.daw21.com
    AuthType Digest
    AuthName "informatica"
    AuthDigestProvider file
    AuthUserFile /etc/apache2/digest
    Require user primo1 primo2
</Directory>
```

Ten en cuenta que en la directiva AuthName tienes que poner lo mismo que pusiste en el dominio o “realm”.

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

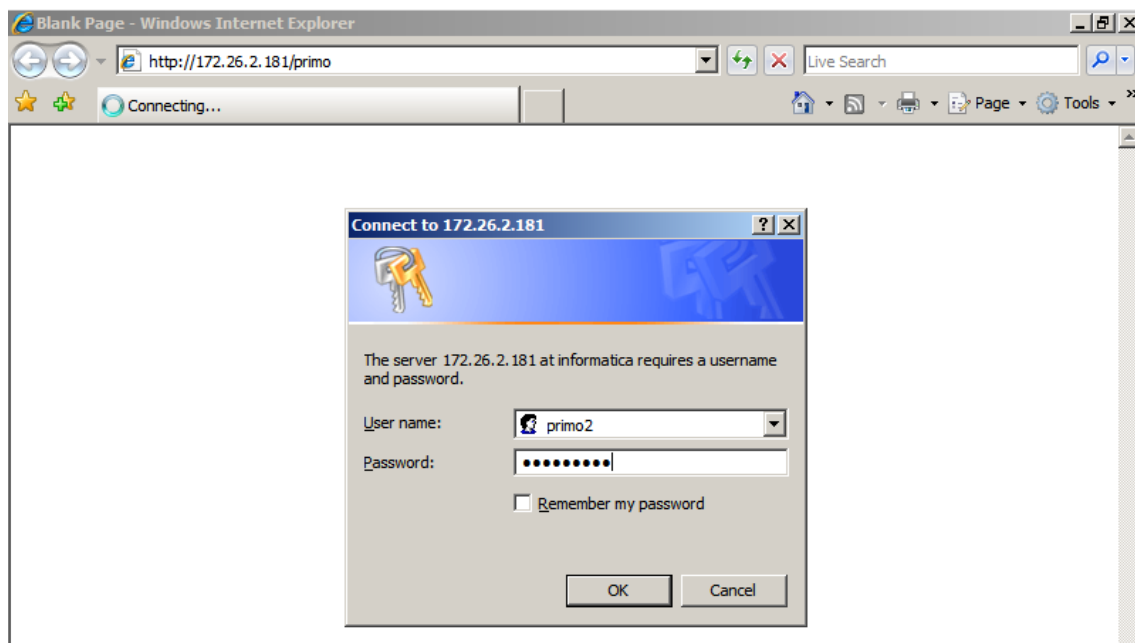
PASO 6) Abre un navegador desde tu máquina física y accede al recurso /primo como usuario primo1.





Hola soy tu primo

PASO 7) Abre un navegador desde tu servidor Windows 2008 y accede al recurso /primo como usuario primo2.



C) Ficheros .htaccess.

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

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.

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

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

PASO 4) Inicia sesión con el nuevo usuario useraccess.

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

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

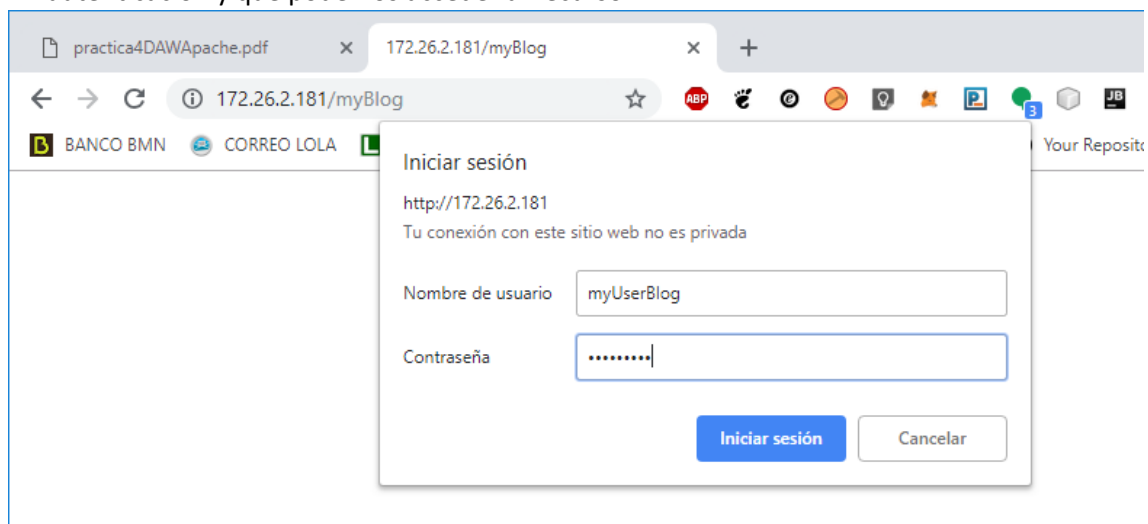
```
soares_teixeira@servidorLinux21:/$ sudo htdigest -c /home/useraccess/myBlog/.htdigest informatica myUserBlog
Adding password for myUserBlog in realm informatica.
New password:
Re-type new password:
soares_teixeira@servidorLinux21:/$
```

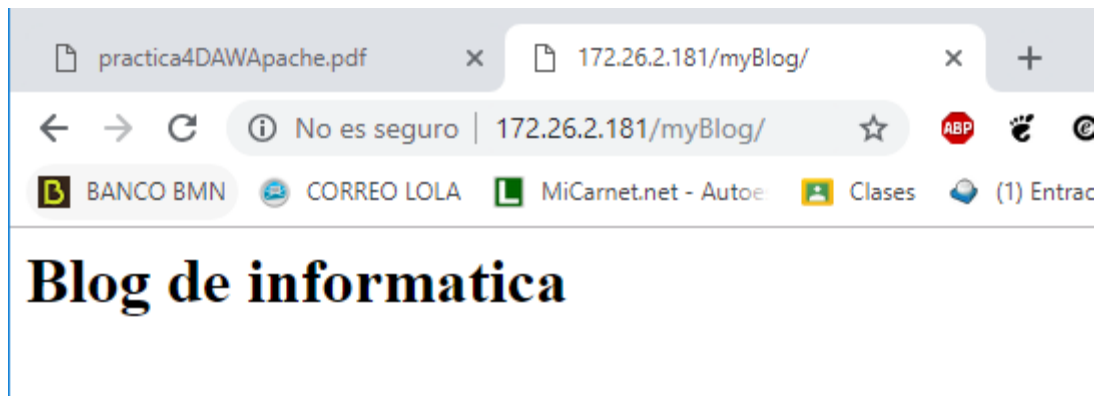
PASO 7) Ahora tendremos que crear el fichero .htaccess (también dentro de myBlog).

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

```
Options Indexes FollowSymLinks Multiviews
Order allow,deny
allow from 172.26.2.121
AuthType Digest
AuthName "informatica"
AuthDigestProvider file
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.





D) Ficheros de registros (logs).

Toma una captura de los pasos 2 y 3

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

1. ¿qué directiva marca la ruta del archivo de los errores?

La Directiva es ErrorLog.

2. ¿Cuál es el fichero de logs de errores? ¿qué nivel de prioridad tiene?

El fichero es: error.log

3. ¿qué directiva marca la ruta del archivo de los accesos?

El nivel es WARN.

4. ¿Cuál es el fichero de logs de accesos?

El fichero es: Acces.log

PASO 2) Consulta el log de errores

```
[Mon Dec 10 12:27:35.939983 2018] [mpm_event:notice] [pid 1858:tid 140283047454592] AH00489: Apache/2.4.7 (Ubuntu) config
resuming normal operations
[Mon Dec 10 12:27:35.940059 2018] [core:notice] [pid 1858:tid 140283047454592] AH00094: Command line: '/usr/sbin/apache2
[Mon Dec 10 14:02:26.600611 2018] [mpm_event:notice] [pid 1858:tid 140283047454592] AH00491: caught SIGTERM, shutting do
[Fri Dec 14 08:49:35.616922 2018] [mpm_event:notice] [pid 951:tid 139887587239808] AH00489: Apache/2.4.7 (Ubuntu) config
resuming normal operations
[Fri Dec 14 08:49:35.629903 2018] [core:notice] [pid 951:tid 139887587239808] AH00094: Command line: '/usr/sbin/apache2
[Fri Dec 14 12:34:27.218707 2018] [mpm_event:notice] [pid 951:tid 139887587239808] AH00491: caught SIGTERM, shutting do
[Fri Dec 14 12:34:28.310953 2018] [mpm_event:notice] [pid 1700:tid 139764302190464] AH00489: Apache/2.4.7 (Ubuntu) config
resuming normal operations
[Fri Dec 14 12:34:28.311013 2018] [core:notice] [pid 1700:tid 139764302190464] AH00094: Command line: '/usr/sbin/apache2
[Fri Dec 14 12:38:17.574804 2018] [mpm_event:notice] [pid 1700:tid 139764302190464] AH00491: caught SIGTERM, shutting do
[Fri Dec 14 12:38:18.666659 2018] [mpm_event:notice] [pid 1807:tid 140284918134656] AH00489: Apache/2.4.7 (Ubuntu) config
resuming normal operations
```

PASO 3) Consulta el log de accesos

```
172.26.2.121 - - [10/Dec/2018:12:33:07 +0100] "GET / HTTP/1.1" 200 3594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) f
it/537.36 (KHTML, like Gecko) Chrome/70.0.3538.110 Safari/537.36"
172.26.2.121 - - [10/Dec/2018:12:33:07 +0100] "GET /icons/ubuntu-logo.png HTTP/1.1" 200 3688 "http://172.26.2.181/" "Moz
(Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.110 Safari/537.36"
172.26.2.121 - - [10/Dec/2018:12:33:08 +0100] "GET /favicon.ico HTTP/1.1" 404 501 "http://172.26.2.181/" "Mozilla/5.0 (
T 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.110 Safari/537.36"
172.26.2.121 - - [10/Dec/2018:12:36:27 +0100] "GET / HTTP/1.1" 200 3594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) f
it/537.36 (KHTML, like Gecko) Chrome/70.0.3538.110 Safari/537.36"
172.26.2.121 - - [10/Dec/2018:12:36:27 +0100] "GET /icons/ubuntu-logo.png HTTP/1.1" 200 3688 "http://servidorlinux21.daw
"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.110 Safari/537.36"
172.26.2.121 - - [10/Dec/2018:12:36:28 +0100] "GET /favicon.ico HTTP/1.1" 404 514 "http://servidorlinux21.daw21.com/" "M
0.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.110 Safari/537.36"
```

E) Módulos status e info.

Toma una captura de los pasos 2 y 4, 6 y 8.

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

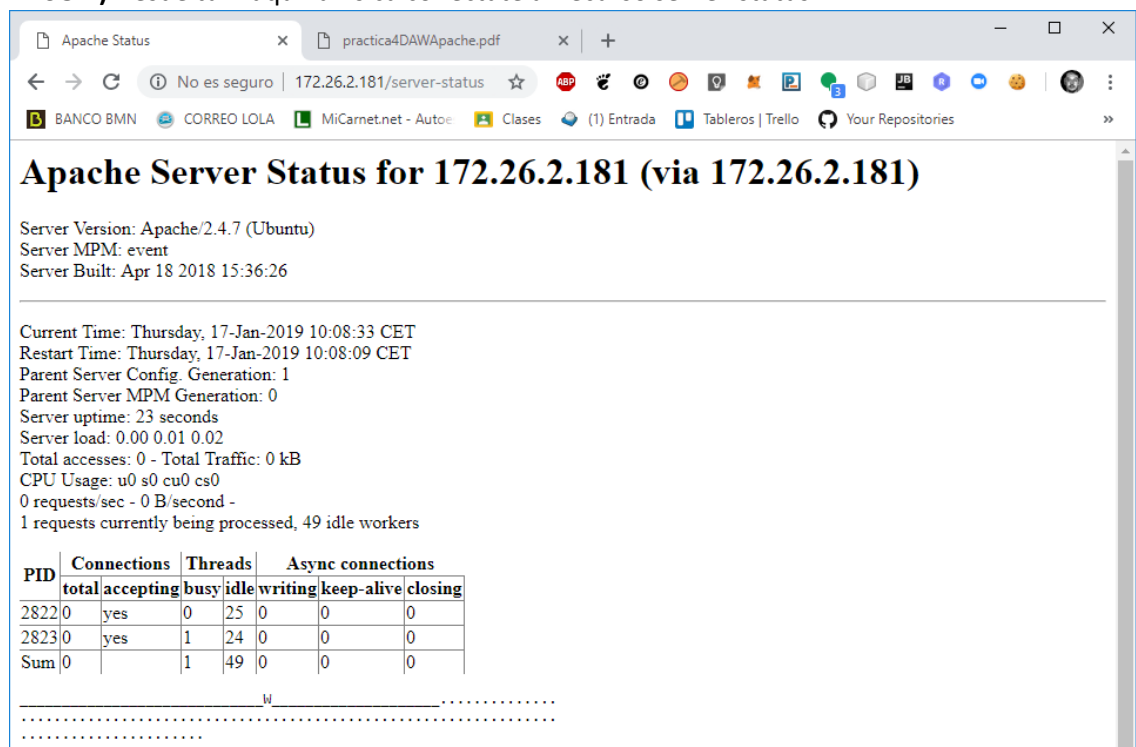
Habilito:

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

```
<Location /server-status>
    SetHandler server-status
    #Require local
    Require ip 172.26.2.121
</Location>
```

PASO 3) Reinicia el servidor para aplicar los cambios.

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



Apache Server Status for 172.26.2.181 (via 172.26.2.181)

Server Version: Apache/2.4.7 (Ubuntu)
Server MPM: event
Server Built: Apr 18 2018 15:36:26

Current Time: Thursday, 17-Jan-2019 10:08:33 CET
Restart Time: Thursday, 17-Jan-2019 10:08:09 CET
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 23 seconds
Server load: 0.00 0.01 0.02
Total accesses: 0 - Total Traffic: 0 kB
CPU Usage: u0 s0 cu0 cs0
0 requests/sec - 0 B/second -
1 requests currently being processed, 49 idle workers

| PID | Connections | | Threads | | Async connections | | |
|------|-------------|-----------|---------|------|-------------------|------------|---------|
| | total | accepting | busy | idle | writing | keep-alive | closing |
| 2822 | 0 | yes | 0 | 25 | 0 | 0 | 0 |
| 2823 | 0 | yes | 1 | 24 | 0 | 0 | 0 |
| Sum | 0 | | 1 | 49 | 0 | 0 | 0 |

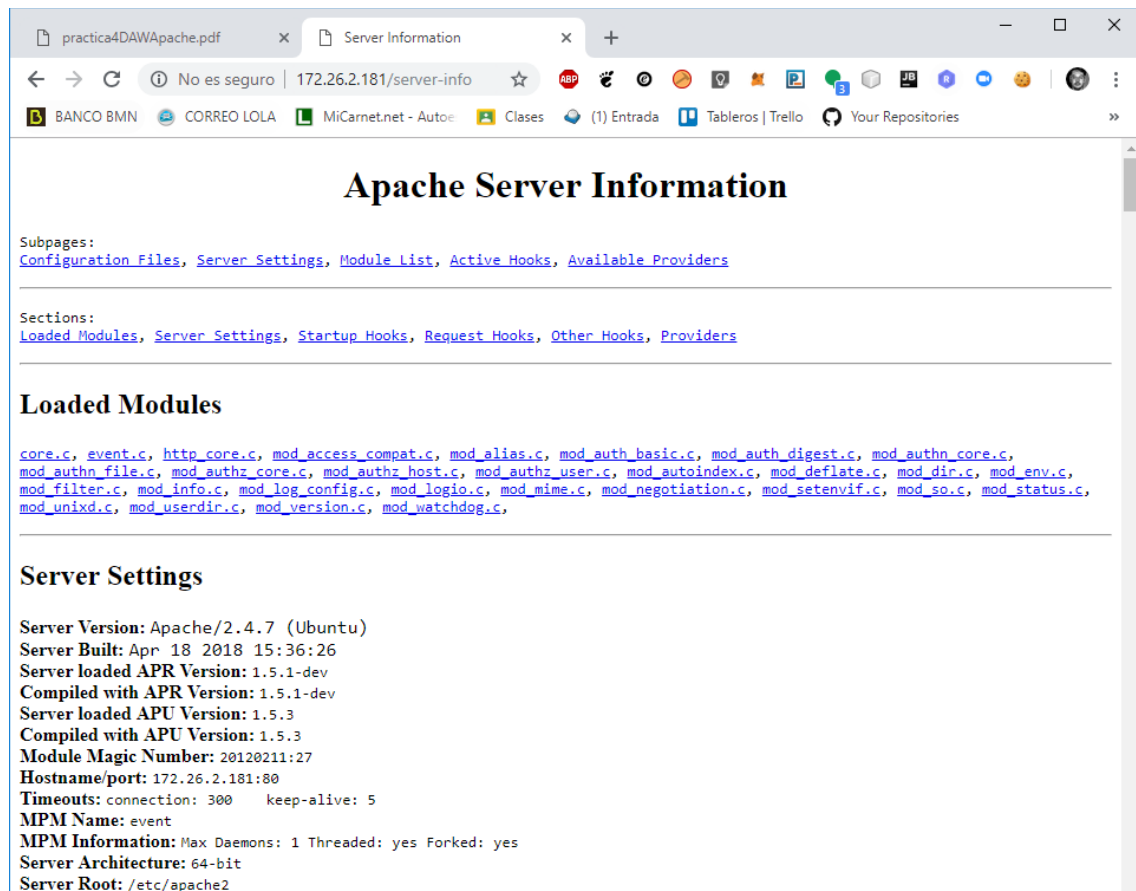
PASO 5) En tu servidor Linux, habilita el módulo info.

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

```
# Allow remote server configuration reports, with the URL of
# http://servername/server-info (requires that mod_info.c be loaded).
# Uncomment and change the "192.0.2.0/24" to allow access from other hosts.
#
<Location /server-info>
    SetHandler server-info
    #Require local
    Require ip 172.26.2.121
</Location>
```

PASO 7) Reinicia el servidor para aplicar los cambios.

PASO 8) Desde tu máquina física conéctate al recurso server-info
Consulta el fichero server-info.



practica4DAWAapache.pdf x Server Information x +

No es seguro | 172.26.2.181/server-info

BANCO BMN CORREO LOLA MiCarnet.net - Autoe Clases (1) Entrada Tableros | Trello Your Repositories

Apache Server Information

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

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

Loaded Modules

[core.c](#), [event.c](#), [http_core.c](#), [mod_access_compat.c](#), [mod_alias.c](#), [mod_auth_basic.c](#), [mod_auth_digest.c](#), [mod_authn_core.c](#), [mod_authn_file.c](#), [mod_authz_core.c](#), [mod_authz_host.c](#), [mod_authz_user.c](#), [mod_autoindex.c](#), [mod_deflate.c](#), [mod_dir.c](#), [mod_env.c](#), [mod_filter.c](#), [mod_info.c](#), [mod_log_config.c](#), [mod_logio.c](#), [mod_mime.c](#), [mod_negotiation.c](#), [mod_setenvif.c](#), [mod_so.c](#), [mod_status.c](#), [mod_unixd.c](#), [mod_userdir.c](#), [mod_version.c](#), [mod_watchdog.c](#),

Server Settings

Server Version: Apache/2.4.7 (Ubuntu)
Server Built: Apr 18 2018 15:36:26
Server loaded APR Version: 1.5.1-dev
Compiled with APR Version: 1.5.1-dev
Server loaded APU Version: 1.5.3
Compiled with APU Version: 1.5.3
Module Magic Number: 20120211:27
Hostname/port: 172.26.2.181:80
Timeouts: connection: 300 keep-alive: 5
MPM Name: event
MPM Information: Max Daemons: 1 Threaded: yes Forked: yes
Server Architecture: 64-bit
Server Root: /etc/apache2

¿tienes cargado el módulo mod_mime?

```
196: AddCharset UTF-32 .utf32
197: AddCharset UTF-32BE .utf32be
198: AddCharset UTF-32LE .utf32le
```

¿en caso que lo tuvieras, tiene el módulo cargada la configuración de caracteres UTF-32?

F) Webalizer.

Toma una captura de los pasos 2 y 5.

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?
 - Coge los datos del Access.log.
- ¿es correcta la ruta y el nombre del fichero? Si no es así, modifícala.
 - La ruta si, el nombre no.

```
# 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
```

PASO 3) La instalación también implica la creación del recurso que se servirá desde el navegador.

¿Dónde está este fichero?

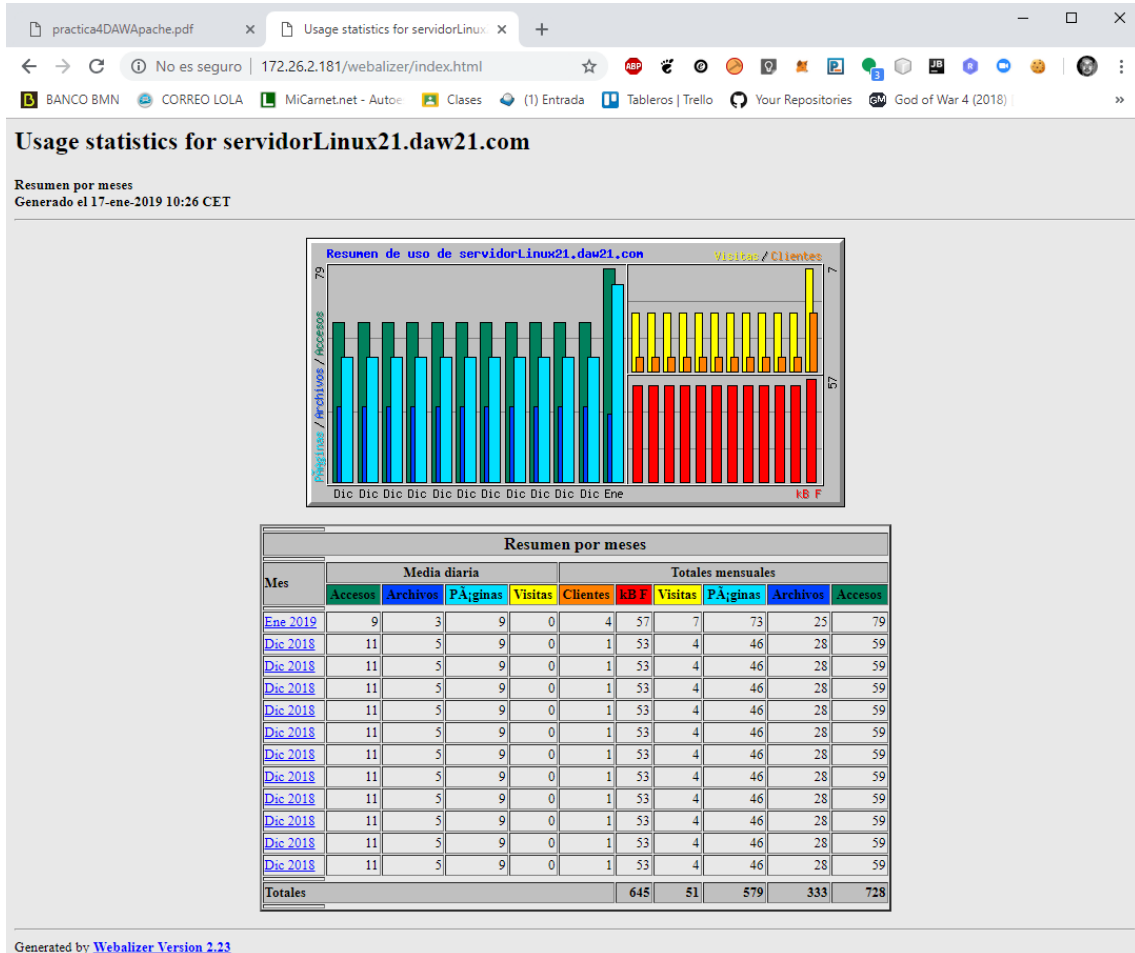
- Esta en var/www/html/webalizer

¿Es correcta la ubicación para servirlo? Si no es así, muévelo a la ubicación correcta.

- No es correcta como viene en el servidor. Lo modifíco.
- /var/www/html/webalizer

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.



G) Alojamiento virtual de sitios web en Linux.

Toma una captura de los pasos 1, 7, 8, 9, 11, 13 y 16.

PASO 1) Lo primero es que se puedan resolver estos nombres a partir de la IP de nuestro servidor Linux (asociar estos nombres con la dirección IP del servidor Linux). Tendremos que crear en nuestro servidor DNS (en el Windows Server) los registros correspondientes (tipo Address).

| | | |
|----------|----------|--------------|
| miTienda | Host (A) | 172.26.2.181 |
| miWiki | Host (A) | 172.26.2.181 |

PASO 2) Deshabilita el servidor virtual por defecto 000-default.conf (una vez hecho, verifica que el archivo ya no está en sites-enabled).

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

PASO 5) Crea el directorio /var/www/html/miTienda y dentro de éste el archivo index.html con el contenido que quieras.

PASO 6) Crea el directorio accesoClientes dentro de miTienda. Dentro de éste, el archivo accesoClientes.html con el contenido que quieras.

PASO 7) Crea el fichero /etc/apache2/miTienda.digest y añade el usuario cliente al dominio tienda (revisa el uso de digest que usamos en puntos anteriores).

```
soares_teixeira@servidorLinux21:/etc/apache2$ sudo htdigest /etc/apache2/miTienda.digest cliente cliente
Adding user cliente in realm cliente
New password:
Re-type new password:
soares_teixeira@servidorLinux21:/etc/apache2$ _
```

PASO 8) En este punto es donde crearemos el servidor virtual. Si te fijas, la estructura es la misma que la del servidor virtual por defecto 000-default.conf, pero las directivas ServerName y DocumentRoot son distintas.

Crea el fichero /etc/apache2/sites-available/miTienda.conf con las siguientes directivas:

```
<VirtualHost *:80>
    ServerName miTienda.daw21.com
    DocumentRoot /var/www/html/miTienda

    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>

    <Directory /var/www/html/miTienda>
        DirectoryIndex index.html
        Options FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    <Directory /var/www/html/miTienda/accesoClientes>
        Options Indexes FollowSymLinks MultiViews
        AuthType Digest
        AuthName "tienda"
        AuthUserFile /etc/apache2/miTienda.digest
        Require user cliente
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/miTienda.error.log
    LogLevel warn
    CustomLog ${APACHE_LOG_DIR}/miTienda.access.log combined
</VirtualHost>
```

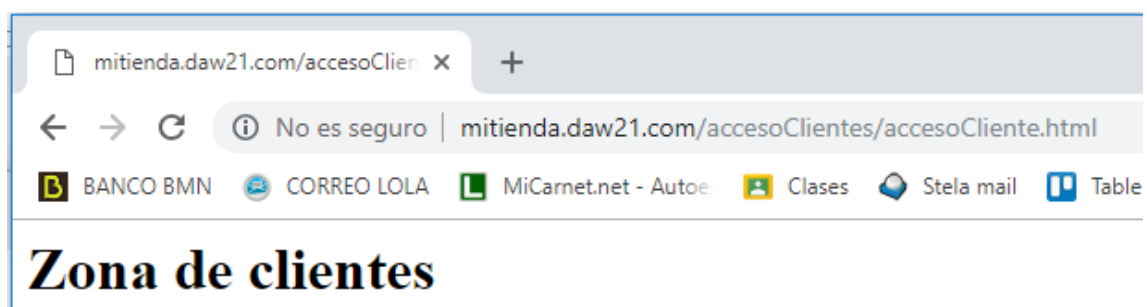
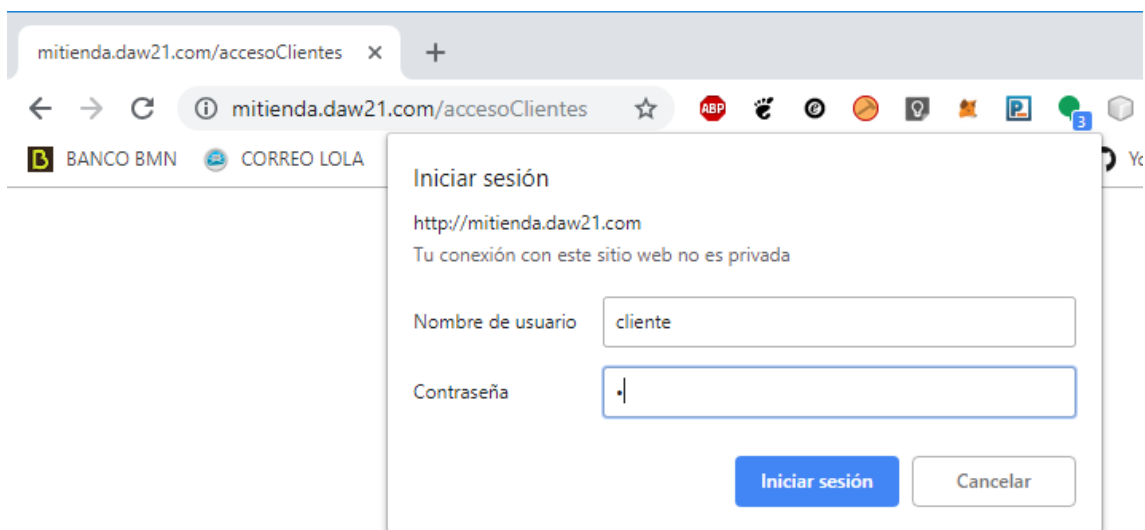
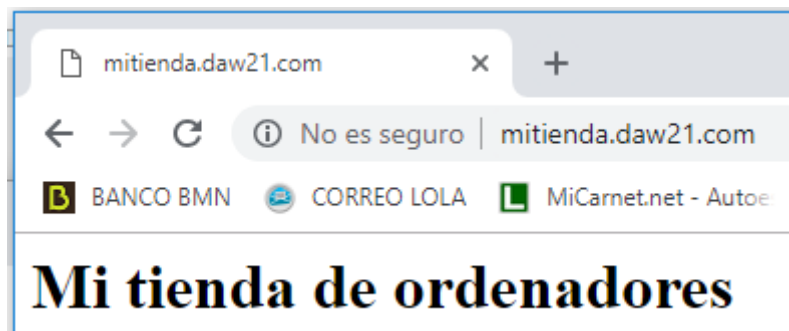
PASO 9) Habilita el servidor virtual miTienda y verifica que se ha creado el enlace en sites-enabled.

```
soares_teixeira@servidorLinux21:/etc/apache2$ ls sites-enabled/
miTienda.conf
soares_teixeira@servidorLinux21:/etc/apache2$ _
```

```
soares_teixeira@servidorLinux21:/etc/apache2$ ls sites-enabled/
miTienda.conf
soares_teixeira@servidorLinux21:/etc/apache2$ _
```

PASO 10) Reinicia el servidor para que los cambios tengan efecto.

PASO 11) Accede desde tu máquina física a miTienda.dawXX.com y a miTienda.dawXX.com/accesoClientes.



PASO 12) Crea el directorio `/var/www/html/miWiki` y dentro de éste el archivo `principal.html` con el contenido que quieras.

PASO 13) Crea el fichero `/etc/apache2/sites-available/miWiki.conf`, guíate por el fichero del servidor virtual que hemos creado anteriormente `miTienda.conf`

```

<VirtualHost *:80>
    ServerName miWiki.daw21.com
    DocumentRoot /var/www/html/miWiki

    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>

    <Directory /var/www/html/miWiki>
        DirectoryIndex principal.html
        Options FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/miWiki.error.log
    LogLevel warn
    CustomLog ${APACHE_LOG_DIR}/miWiki.access.log combined
</VirtualHost>

```

PASO 14) Habilita el servidor virtual miWiki y verifica que se ha creado el enlace en sites-enabled.

PASO 15) Reinicia el servidor para que los cambios tengan efecto.

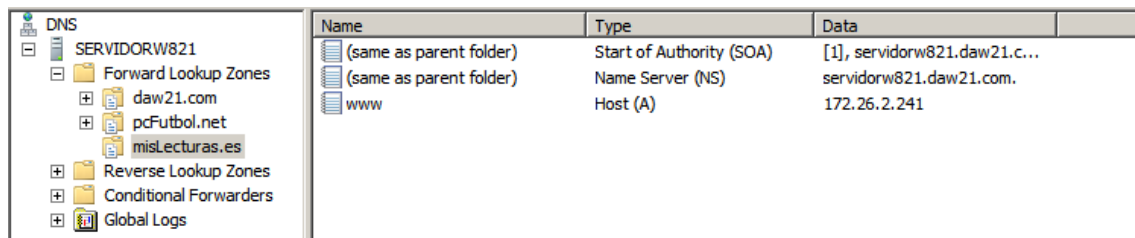
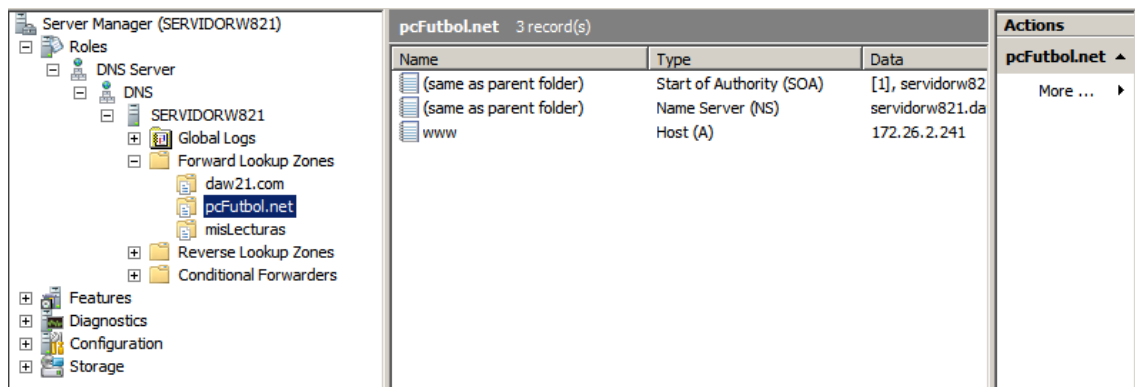
PASO 16) Accede desde tu máquina física a miWiki.dawXX.com, debe servir el archivo principal.html



H) Alojamiento virtual de sitios web en Windows.

Toma instantáneas de los puntos 1,2,3,4,5 y 7.

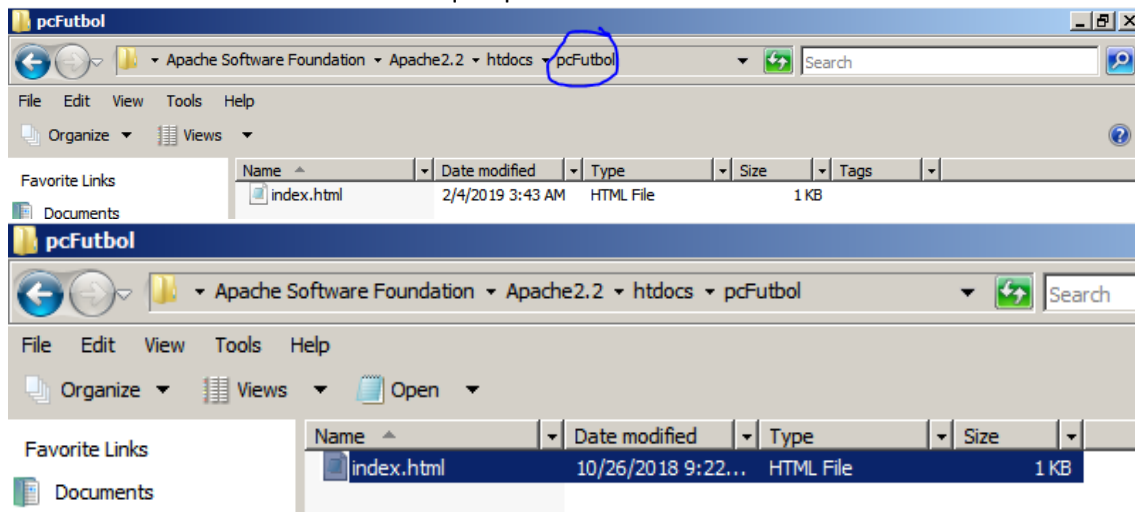
PASO 1: Lo primero es que se puedan resolver estos nombres a partir de la IP de nuestro servidor Windows (asociar estos nombres con la dirección IP del servidor Windows). Tendremos que crear en nuestro servidor DNS (en el Windows Server) dos nuevas zonas.



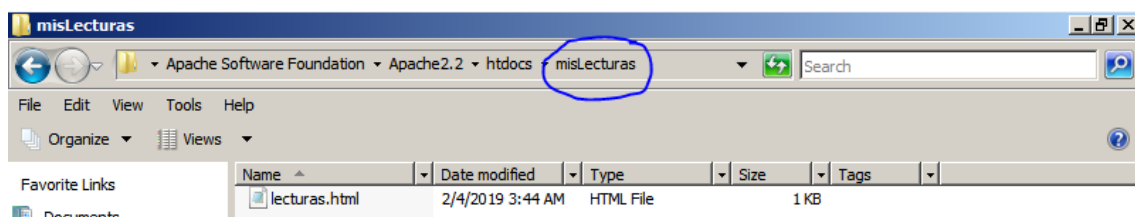
PASO 2) Ahora habilitaremos el módulo para el uso de servidores virtuales, quitando el comentario de la directiva include para httpd-vhosts.conf dentro del archivo httpd.conf

```
# virtual hosts
Include conf/extra/httpd-vhosts.conf
```

PASO 3) Dentro de la carpeta htdocs, crearemos la carpeta pcFutbol y dentro de ésta, el archivo index.html con el contenido que quieras.



PASO 4) Dentro de la carpeta htdocs, crearemos la carpeta misLecturas. Dentro de ésta, el archivo lecturas.html con el contenido que quieras.



PASO 5) Edita el fichero httpd-vhosts.conf e incluye las directivas para poder servir los contenidos creados anteriormente (te muestro el que necesitas para pcFutbol y tienes que construir el otro a partir de éste).

```
<VirtualHost *:80>
    DocumentRoot "c:/Program Files (x86)/Apache Software Foundation/Apache2.2/htdocs/pcFutbol"
    ServerName www.pcFutbol.net

    <Directory />
        DirectoryIndex index.html
        Options Indexes FollowSymLinks
        AllowOverride None
        Order deny,allow
        Allow from all
    </Directory>

    ErrorLog "logs/pcFutbol.error.log"
    CustomLog "logs/pcFutbol.access.log" common
</VirtualHost>

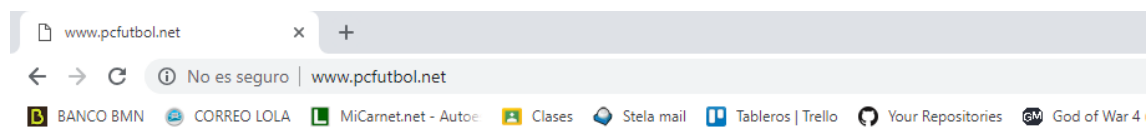
<VirtualHost *:80>
    DocumentRoot "c:/Program Files (x86)/Apache Software Foundation/Apache2.2/htdocs/misLecturas"
    ServerName www.misLecturas.es

    <Directory />
        DirectoryIndex lecturas.html
        Options Indexes FollowSymLinks
        AllowOverride None
        Order deny,allow
        Allow from all
    </Directory>

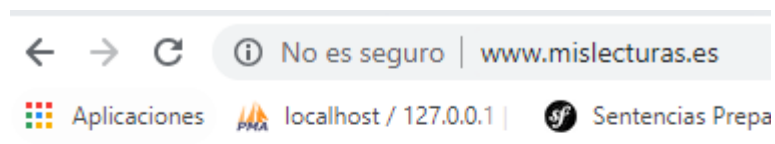
    ErrorLog "logs/misLecturas.error.log"
    CustomLog "logs/misLecturas.access.log" common
</VirtualHost>
```

PASO 6) Reinicia el servidor para que los cambios tengan efecto.

PASO 7) Desde tu máquina física, accede a www.pcFutbol.net y a www.misLecturas.es.

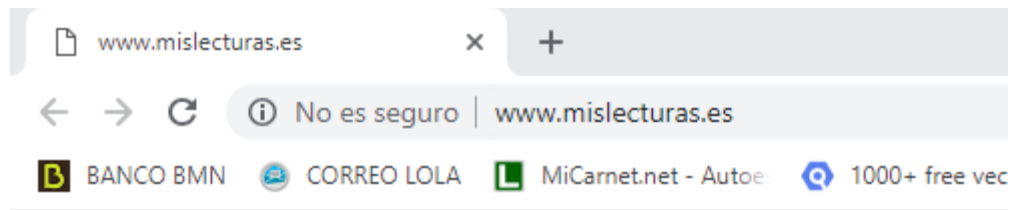


pcFutbol la mejor página de ordenadores y que juegan al futbol



Index of /

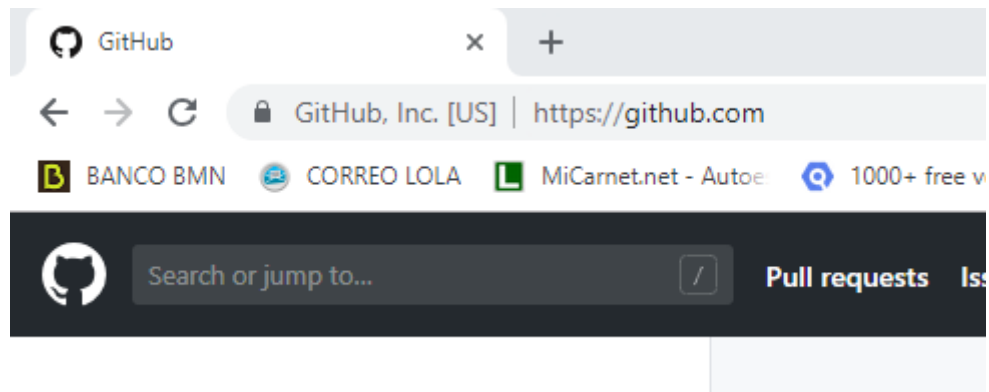
- [lecturas.html.html](#)

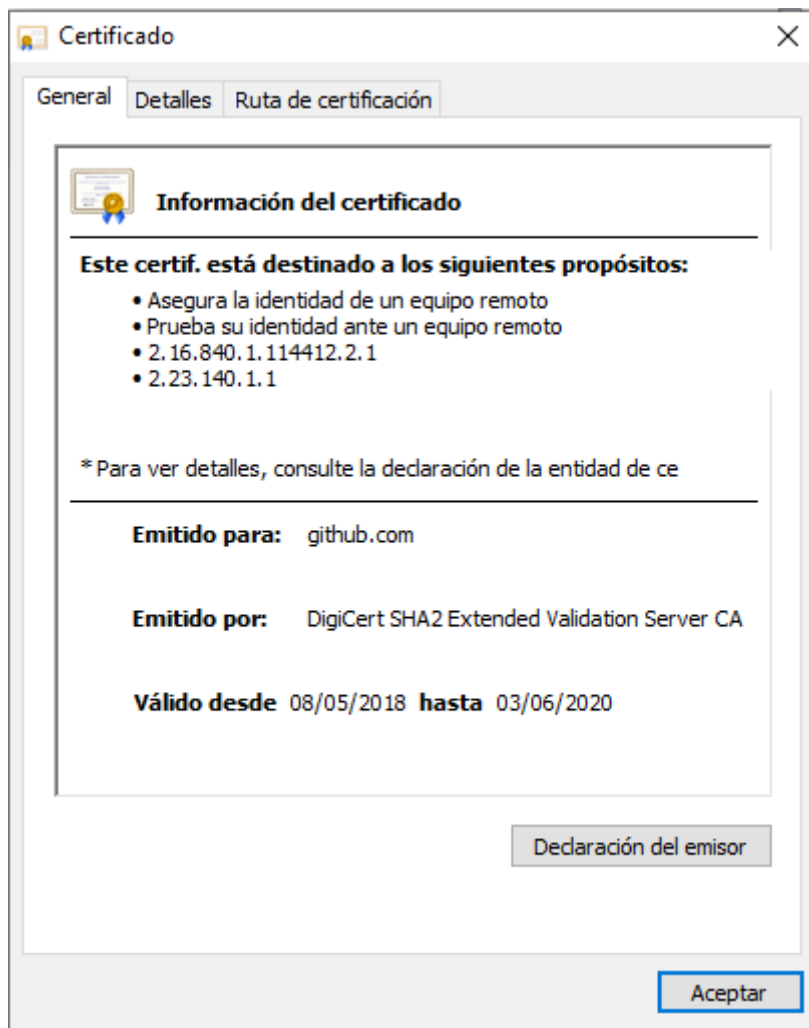


I) HTTPS y Certificados Digitales.

Toma instantáneas de los puntos 1 y del certificado en el paso 3.

PASO 1) Accede desde tu navegador a una página que use el protocolo https.





PASO 2) En la parte izquierda de la barra de direcciones, debe aparecer un candado indicando que la URL es segura. Si pinchas sobre el candado podrás acceder a la información sobre el certificado digital de la misma.

PASO 3) Accede al certificado digital y contesta las siguientes preguntas:

- ¿Qué algoritmo de clave pública usa el certificado?
 - El Algoritmo de clave publica es : ECC.
- ¿Para qué se usa el algoritmo de clave pública?
 - El Algoritmo de clave publica es aquel en el se basa en el uso de una pareja de clave pública y privada de las cuales una se usa para cifrar y otra para descifrar.
- ¿De qué tamaño es la clave pública del certificado?
 - 256 bits
- ¿Qué algoritmo de firma usa el certificado?
 - Sha256.
- ¿Qué longitud tiene?
 - 256 bits.

- ¿Qué autoridad de certificación ha firmado el certificado?
 - Google Internet Authority G3.

J) Servidor virtual HTTPS por defecto en Linux.

Toma instantáneas del punto 4,6,8,10,11,12,13

PASO 1) Habilita el servidor virtual por defecto en apache. Verifica que se ha habilitado correctamente.

PASO 2) Deshabilita los servidores virtuales creados en puntos anteriores.

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

PASO 4) Habilita el módulo ssl para poder hacer uso de servidores virtuales seguros.

```
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
    service apache2 restart
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$ _
```

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

PASO 6) Consulta el fichero port.conf en etc/apache2/.

- ¿En qué puerto escucha el servidor por defecto?
 - Se ESCUCHA POR EL 80.
- ¿Qué puerto está a la escucha si habilitamos el módulo ssl?
 - 443
- ¿Qué directiva lo marca?
 - ifModule ssl_module>

```
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>
```

PASO 7) Verifica que los puertos para el servidor apache están realmente a la escucha mediante el comando `netstat -ltn`

PASO 8) Accede al directorio `sites-available` y realiza un listado,

```
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$ ls -lha
total 28K
drwxr-xr-x 2 root root 4,0K feb  4 12:23 .
drwxr-xr-x 8 root root 4,0K ene 28 12:37 ..
-rw-r--r-- 1 root root 3,3K ene 17 09:28 000-default.conf
-rw-r--r-- 1 root root 6,3K ene  7 2014 default-ssl.conf
-rw-r--r-- 1 root root  691 ene 28 13:01 miTienda.conf
-rw-r--r-- 1 root root  468 feb  4 12:23 miWiki.conf
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$
```

- ¿qué archivo hay en este directorio para que podamos definir servidores virtuales seguros? Habilitalo.
 - El archivo es `default-ssl.conf`.
 - HABILITADO

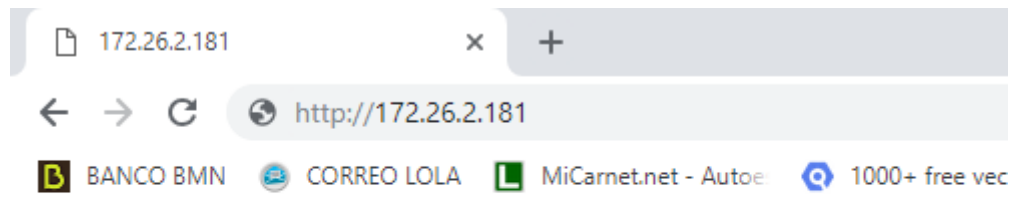
```
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$ ls -lha
total 28K
drwxr-xr-x 2 root root 4,0K feb  4 12:23 .
drwxr-xr-x 8 root root 4,0K ene 28 12:37 ..
-rw-r--r-- 1 root root 3,3K ene 17 09:28 000-default.conf
-rw-r--r-- 1 root root 6,3K ene  7 2014 default-ssl.conf
-rw-r--r-- 1 root root  691 ene 28 13:01 miTienda.conf
-rw-r--r-- 1 root root  468 feb  4 12:23 miWiki.conf
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$
```

PASO 9) Reinicia el servidor.

PASO 10) Abre el archivo indicado en el paso 8.

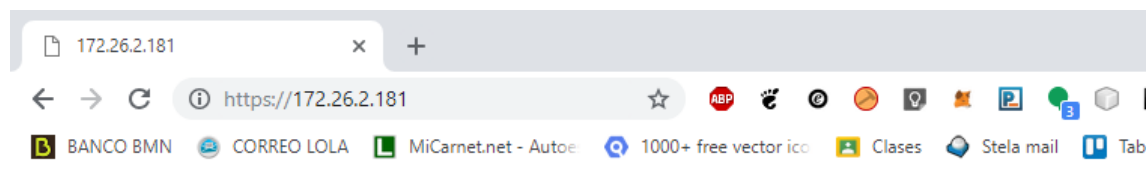
- ¿en qué ruta se deben guardar los certificados SSL en apache?
 - `/etc/ssl/certs`
- ¿y las claves (key)?
 - `/etc/ssl/private`

PASO 11) Desde tu máquina física, abre un navegador y establece una conexión http con el servidor Linux.



Despliegue despliega un pliegue ;)

PASO 12) Establece una conexión https con el servidor Linux.



Este sitio web no puede proporcionar una conexión segura

172.26.2.181 ha enviado una respuesta no válida.

[Prueba a ejecutar Diagnósticos de red de Windows.](#)

ERR_SSL_PROTOCOL_ERROR

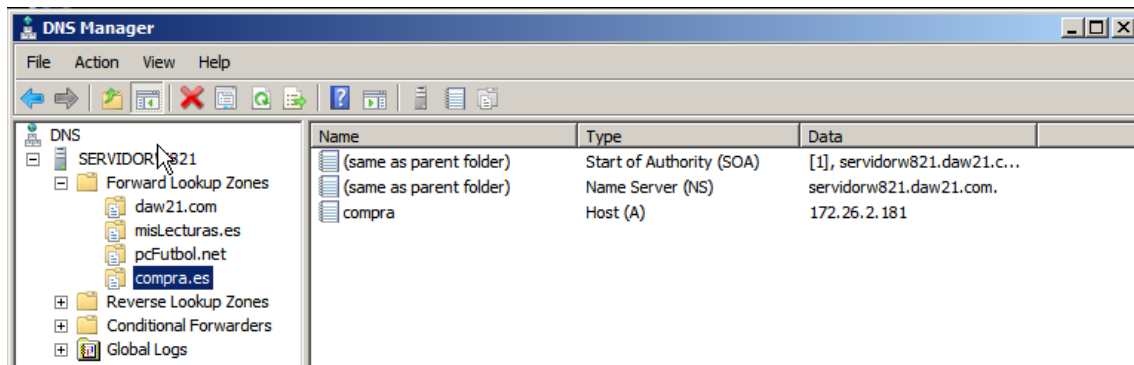
[Volver a cargar](#)

PASO 13) Acepta el acceso. ¿el certificado está expedido por una autoridad certificadora o es autofirmado?

K) Creación de un servidor virtual HTTPS en Linux.

Toma capturas de los puntos 1,3,4,6 y 9

PASO 1) Lo primero es que se puedan resolver el nombre de nuestro nuevo servidor virtual a partir de la IP de nuestro servidor Linux. Tendremos que crear en nuestro servidor DNS (que está en el servidor Windows) un registro para nuestro nuevo servidor virtual.



PASO 2) En el servidor Linux, crea el directorio /compra en /var/www/html/ y crea dentro el fichero index.html con el contenido que quieras.

PASO 3) Vamos a crear un certificado digital autofirmado usando el comando openssl.

- Sitúate en el directorio home del usuario con el que has iniciado sesión.

- Crea una clave privada tipo RSA de 2048 bits usando el siguiente comando:

openssl genrsa -out compra.key 2048

```
soares_teixeira@servidorLinux21:~$ openssl genrsa -out compra.key 2048
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
```

- Genera una solicitud de certificado, a partir de la clave creada en el paso anterior, usando el siguiente comando (inventa los datos introducidos):

openssl req -new -key compra.key -out compra.csr

```
soares_teixeira@servidorLinux21:~$ openssl req -new -key compra.key -out compra.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:ES
State or Province Name (full name) [Some-State]:Sevilla
Locality Name (eg, city) []:Sevilla
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Compras
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:Paulo
Email Address []:pauloxti@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:1
string is too short, it needs to be at least 4 bytes long
A challenge password []:paulo1994
An optional company name []:
soares_teixeira@servidorLinux21:~$
```


Esta solicitud de certificado se podría enviar a una autoridad de certificación para que nos generase un certificado CRT avalado por una entidad certificadora. Nosotros vamos a autofirmar el certificado para generarlo nosotros mismos.

- Crea el certificado digital autofirmado usando la clave privada y la solicitud de certificado generados anteriormente. Usa el siguiente comando:

openssl x509 -req -days 365 -in compra.csr -signkey compra.key -out compra.crt

```
soares_teixeira@servidorLinux21:~$ openssl x509 -req -days 365 -in compra.csr -signkey compra.key -out compra.crt
Signature ok
subject=C=ES/ST=Sevilla/L=Sevilla/O=Compras/CN=Paulo/emailAddress=pauloxti@gmail.com
Getting Private key
soares_teixeira@servidorLinux21:~$
```

PASO 4) Mueve la clave (.key) y el certificado (.crt) a los directorios que utiliza por defecto Apache para guardar los certificados y las claves (paso 10 punto anterior).

```
soares_teixeira@servidorLinux21:~$ openssl x509 -req -days 365 -in compra.csr -signkey compra.key -out compra.crt
Signature ok
subject=C=ES/ST=Sevilla/L=Sevilla/O=Compras/CN=Paulo/emailAddress=pauloxti@gmail.com
Getting Private key
soares_teixeira@servidorLinux21:~$
```

```
soares_teixeira@servidorLinux21:~$ sudo mv compra.crt /etc/ssl/certs/
soares_teixeira@servidorLinux21:~$ sudo mv compra.key /etc/ssl/private/
```

PASO 5) Configura los permisos adecuados de la siguiente manera.

sudo chown root:ssl-cert /etc/.../compra.key

sudo chmod 640 /etc/.../compra.key

sudo chown root:root /etc/.../compra.crt

PASO 6) Crea el fichero compra.conf en el directorio correspondiente dentro de la estructura de apache2. Añade a las directivas SSLCertificateFile y SSLCertificateKeyFile las rutas necesarias de los archivos compra.crt y compra.key

```
<IfModule mod_ssl.c>
<VirtualHost *:443>
    ServerName compra.daw21.com
    DocumentRoot /var/www/html/compra

    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>

    <Directory /var/www/html/compra>
        DirectoryIndex index.html
        Options Indexes FollowSymLinks
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/compra.log
    LogLevel warn
    CustomLog ${APACHE_LOG_DIR}/compra.access.log combined

    SSLEngine on
    SSLCertificateFile /etc/ssl/certs/compra.crt
    SSLCertificateKeyFile /etc/ssl/private/compra.key
</VirtualHost>
</IfModule>
```

PASO 7) Deshabilita el servidor ssl por defecto y habilita compra.conf.

```
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$ sudo a2dissite default-ssl.conf
Site default-ssl disabled.
To activate the new configuration, you need to run:
    service apache2 reload
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$ _
```

```
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$ sudo a2ensite compra.conf
Enabling site compra.
To activate the new configuration, you need to run:
    service apache2 reload
soares_teixeira@servidorLinux21:/etc/apache2/sites-available$
```

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

PASO 9) Desde tu máquina física, abre un navegador y establece una conexión

<https://compra.dawXX.com>, permite en tu navegador el acceso a la página y abre el certificado para ver la información.

- Del acceso inicial cuando aún no has aceptado el certificado
- Del acceso una vez aceptado el certificado.
- Del propio certificado digital cuando accedas a él