

# DESPLIEGUE DE APLICACIONES WEB

## PRÁCTICA 3. ADMINISTRACIÓN DE APACHE

### A) Instalación del servidor web Apache

**Apache** es un servidor web “open source” potente, flexible y adaptado al protocolo HTTP/1.1.

Es un servidor **modular** (permite añadir extensiones de manera sencilla) y **multiplataforma** (Windows, Linux...).

#### A.1) Instalación apache en Linux

**Paso 1)** Después de iniciar sesión en nuestro servidor Linux actualiza el sistema operativo e instala la versión 2 de apache con los siguientes comandos (ten en cuenta que el servidor DNS de tu servidor Linux está alojada en tu máquina Servidor Windows, por lo que para actualizar e instalar tienen que estar ambas máquinas corriendo):

```
Sudo apt-get update  
Sudo apt-get install apache2
```

Una vez instalado el servidor apache se crearán los archivos de configuración del servidor Apache que iremos viendo más adelante.

Las páginas que sirva estarán, por defecto en la carpeta **/var/www/html**, la cual es propiedad del usuario **root** y del grupo **root**.

#### Toma una captura de pantalla de la comprobación anterior.

**Paso 2)** La instalación ha creado un usuario www-data que pertenecerá al grupo www-data. Además, ahora el servidor debería estar a la escucha. Compruébalo con los siguientes comandos:

```
Ps -ef | grep apache  
Netstat -ltn
```

¿Qué puerto es el que está a la escucha para este servidor?

#### Toma capturas de pantallas del paso 2.

**Paso 3)** Para comprobar la versión usaremos el comando `apache2 -v`

Comprueba además el contenido de **/var/www/html/**

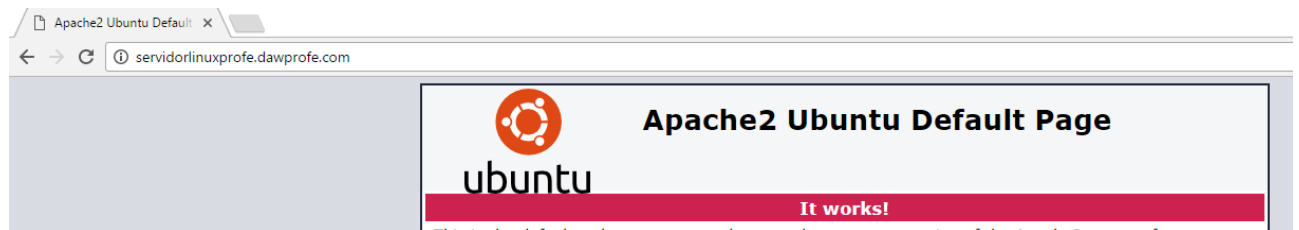
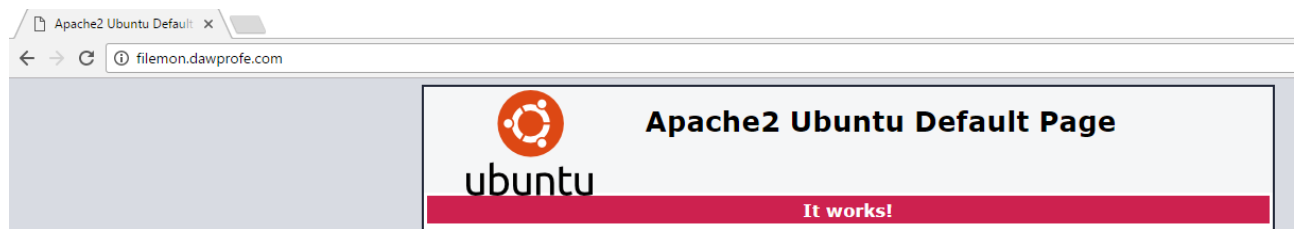
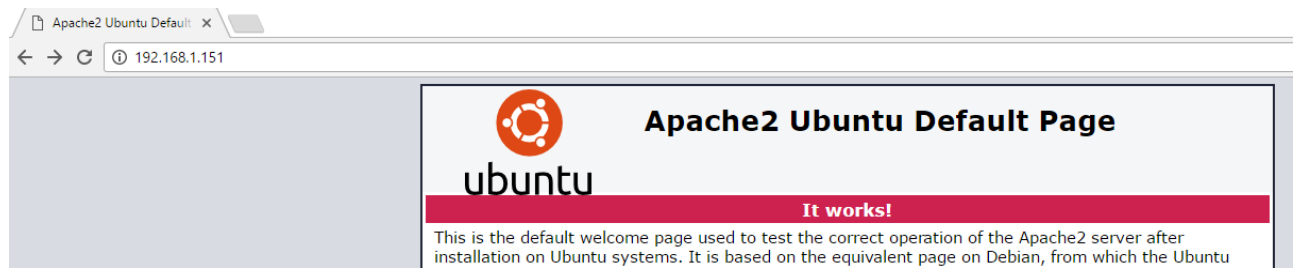
¿Qué archivo(s) está dentro de este directorio? ¿Qué es este archivo?

#### Toma capturas de contenido del archivo que está en /var/www/html

**Paso 4)** Desde nuestra máquina física, vamos a comprobar la conexión con nuestro servidor Apache, abriendo un navegador y accediendo:

- Con la dirección IP de nuestra máquina Linux desde donde sirve contenido Apache.

- Con el nombre de nuestra máquina Linux, o alguno de los alias (para ello, nuestra máquina física debe tener como servidor DNS la IP de nuestra máquina servidor Windows).



¿Qué archivo está sirviendo Apache y en qué carpeta está?

**Toma capturas de cada uno de los accesos desde el navegador al Apache.**

## A.2) Instalación apache en Windows 2008 Server

Ahora vamos a instalar el servidor HTTP Apache en nuestra máquina Servidor Windows.

**Paso 1)** En nuestro servidor Windows 2008, abrimos un navegador (por cuestiones de seguridad, te saldrán ventanas emergentes para bloquear o agregar los sitios a los que accedes desde el navegador que tendrás que cerrar) y vamos al enlace:

<https://archive.apache.org/dist/httpd/binaries/win32/>

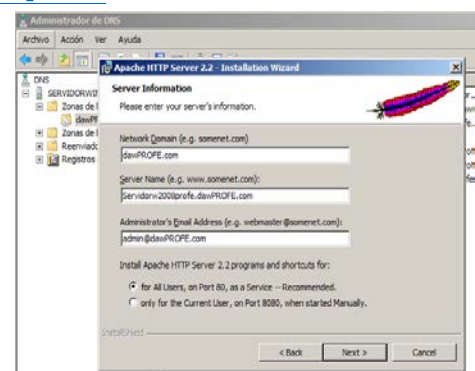
Descarga el archivo: [httpd-2.2.25-win32-x86-openssl-0.9.8y.msi](http://httpd-2.2.25-win32-x86-openssl-0.9.8y.msi) e instálalo.

Durante la instalación configura las opciones tal y como se muestra en la siguiente imagen (adaptado al nombre de tu máquina y tu dominio):

**Toma una captura con tu configuración.**

Elige la **opción Custom** y pulsa **Next**, **Next** e **Instalar**.

Una vez terminada, tu servidor debe estar a la escucha



**Paso 2)** Una vez instalado, comprueba que tu servidor Apache está a la escucha usando en una consola el comando: **netstat -a -p TCP -n**

**Toma una captura de esta comprobación.**

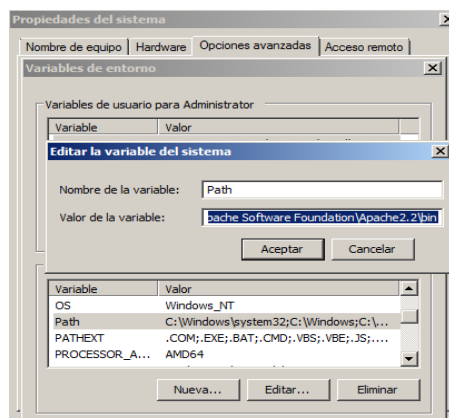
**Paso 3)** Comprueba el contenido del directorio **C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\htdocs** y comprueba el contenido del fichero index.

**Paso 4)** Ahora mismo no se permitirán conexiones a nuestro servidor Apache porque el Firewall tiene bloqueado el acceso. Tendremos que habilitar la regla que nos permita la conexión:

- Accede al Firewall de Windows
- Pincha en Agregar programa
- Pincha en Examina y selecciona:  
**C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\bin\httpd.exe**
- *Acepta los cambios.*

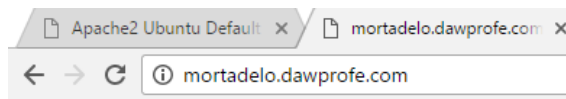
Para utilizar las aplicaciones de Apache desde cualquier localización en la línea de comandos podemos añadir la ruta de los binarios de Apache en la variable de entorno PATH:

- Accede a Inicio, Panel de control, Sistema, Configuración Avanzada del Sistema, Variables de Entorno.
- En Variables de sistema selecciona la variable Path y pincha en Editar.
- Añade la ruta **C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\bin**
- Acepta los cambios



**Paso 5)** Vamos a comprobar la conexión con nuestro servidor Apache instalado en nuestra máquina Servidor Windows 2008:

- **Desde nuestra propia máquina servidor Windows** accediendo con el navegador a la dirección de loopback 127.0.0.1
- **Desde nuestra máquina física con la dirección IP** de nuestra máquina Servidor Windows 2008 desde donde sirve contenido Apache.
- **Desde nuestra máquina física con el nombre de nuestra máquina ServidorWindows, o alguno de los alias** (para ello, nuestra máquina física debe tener como servidor DNS la IP de nuestra máquina servidor Windows).



**It works!**

**Toma capturas de pantallas de cada acceso pedido en el punto anterior.**

## **B) Ficheros de configuración y directivas en Apache para Linux**

**Los ficheros de configuración de Apache en Linux** se encuentran en la carpeta **/etc/apache2**.

Estos ficheros son:

- apache2.conf: es el fichero de configuración principal. Contiene un conjunto de directivas que determinan el comportamiento del servidor.
- ports.conf: en él se definen las IPs y puertos donde escucha el servidor.

Dentro de los ficheros de configuración existen una serie de **directivas** para modelar el comportamiento de nuestro servidor Apache a la hora de servir contenidos. Puedes consultarlas en el enlace:

<http://luismido.wikidot.com/directivas-basicas-de-configuracion-de-apache-2>  
(te será útil para desarrollar esta práctica).

En este punto vamos a analizar los ficheros de configuración del servidor Apache que hemos instalado en nuestra máquina Linux.

### **B.1) Análisis de los ficheros `apache2.conf`, `port.conf` y `000-default.conf`**

**Paso 1)** Cambia a usuario root para poder manejar estos archivos.

**Paso 2)** Haz un listado del contenido que hay en el directorio `/etc/apache2/`.

**Paso 3)** Abre el fichero `/etc/apache2/apache2.conf`. Vamos a analizar su configuración:

- ¿Cuál es el Path de la directiva `ServerRoot`?
- ¿Permite conexiones persistentes por defecto (significa que todas las conexiones de un usuario se sirven en la misma conexión TCP)? ¿Qué directiva define este comportamiento?
- ¿Cuál es el tiempo de espera para enviar/recibir peticiones del servidor? ¿Qué directiva marca este tiempo de espera?
- ¿En qué fichero se registran los errores? ¿Qué directiva lo define?

- ¿Para qué sirve la directiva LogLevel?

**Toma capturas de pantallas de cada una de las directivas por las que se pregunta en el paso 3.**

**Sites-available** contiene por defecto el archivo de configuración de servidores virtuales.

Recuerda que una misma dirección IP puede servir, actuando con diferentes identificadores (usando alias en el servidor DNS) y por tanto distintos contenidos, como si se estuvieran sirviendo desde servidores físicos distintos.

**Sites-enabled** sólo contiene enlaces simbólicos a los archivos que hay en sites-available. Aquellos archivos de sites-available que estén enlazados en el directorio sites-enabled estarán habilitados.

**Paso 4)** Accede al directorio **/etc/apache2/sites-available** y comprueba que está creado el archivo **000-default.conf** (es el archivo que contiene la configuración del servidor virtual por defecto). Abre el archivo para ver el contenido.

- ¿Qué directiva indica el directorio desde donde se sirven los contenidos del servidor virtual? ¿Cuál es la ruta de ese directorio?

**Paso 5)** Accede a **/etc/apache2/sites-enabled** y comprueba que existe el fichero **000-default.conf**. Comprueba que es un enlace simbólico al archivo del punto anterior.

**Paso 6)** Abre el archivo **/etc/apache2/ports.conf**. ¿qué puerto es el que está a la escucha? Si usáramos conexión segura, ¿qué otro puerto estaría a la escucha?

**Toma capturas de pantallas de los pasos 4 5 y 6.**

Para indicar desde qué ubicación se sirven los contenidos se usa la directiva:

**<Directory path>**

...

**</Directory>**

Esta directiva marca como se servirá el contenido que hay en el directorio indicado por **path**.

```
<Directory /var/www/>
    Order allow,deny
    allow from all
    Options None
</Directory>
```

**Paso 7)** Consulta el fichero **/etc/apache2/apache2.conf** y observa como la directiva contenedora que determina como Apache sirve el contenido del directorio **/var/www**.

**Toma capturas de pantallas del paso 7**

## C) Ficheros de configuración y directivas en Apache para Windows2008xx

El fichero de configuración de Apache en Windows se encuentra en la carpeta *C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\conf\* y es **httpd.conf**.

El contenido de este fichero incluye:

- **Puerto a la escucha** para el servidor Apache
- Carga de módulos **LoadModule** (para habilitarlos sólo hay que descomentarlos (quitándole #) y para deshabilitarlos sólo hay que comentarlos (poniéndole #).
- Directiva con la ruta de la carpeta desde donde se servirán los contenidos (**DocumentRoot**).
- Directivas **<Directory></Directory>** indicando las opciones de servicio de cada contenido.
- Directivas **<IfModule></IfModule>** indicando las opciones para los módulos que se usen en el servidor.
- Directivas **include** con la ubicación de los módulos para su uso (para habilitarlos sólo hay que descomentarlos (quitándole #) y para deshabilitarlos sólo hay que comentarlos (poniéndole #).

En este punto vamos a analizar el fichero de configuración del servidor Apache que hemos instalado en nuestra máquina servidor Windows.

### C.1) Análisis del fichero httpd.conf y los ficheros de servidores virtuales

**Paso 1)** Accede al directorio *C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\conf\*

**Paso 2)** Abre el archivo **httpd.conf** y vamos a analizar el contenido:

- ¿Cuál es el path de la directiva **ServerRoot**?
- ¿Qué puerto está a la escucha para el servidor Apache?
- ¿Cuál es el valor de la directiva **DocumentRoot**?
- ¿Desde donde se sirve el contenido según la directiva **Directory**?
- ¿Dónde se guarda el registro de errores según la directiva **ErrorLog**?

Toma capturas de pantallas cada una de las directivas por las que te pregunta el Paso 2.

En Apache para Windows, el servidor virtual por defecto y los que queramos añadir se encuentran en los ficheros **httpd-default** y **httpd-vhosts** respectivamente. Ambos se encuentran en la ruta:

*C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\conf\extra\*

**Paso 3)** Abre los ficheros **httpd-default** y **httpd-vhosts** y observa el contenido de ambos.

Toma capturas de pantallas de cada uno de estos ficheros.

## D) Configuración básica en Apache para Linux

En esta parte de la práctica vamos a añadir distintos contenidos haciendo uso de las directivas **<Directory></Directory>**, así como la configuración para servir estos contenidos a través de directivas como **DirectoryIndex**, **Options Indexes**, **ErrorDocument**, **Alias** y **Redirect**.

### D.1) Ficheros y directorios de prueba.

**Paso 1)** En nuestro Servidor Linux vamos a crear los siguientes directorios y archivos en la carpeta `/var/www/html` (recuerda que es la carpeta desde donde se servirán los contenidos de nuestro servidor Apache).

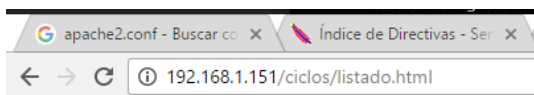
Añade a los archivos contenido HTML (usa encabezados H1), el que quieras, siempre que no hiera la sensibilidad de quien lo lea.

- `/var/www/html/despliegue.html`
- `/var/www/html/fp.html`
- `/var/www/html/ciclos/listado.html`
- `/var/www/html/ciclos/asir.html`
- `/var/www/html/ciclos/daw.html`
- `/var/www/html/ciclos/dam.html`

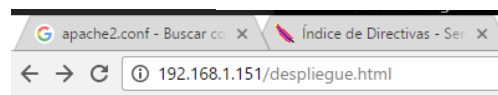
Haz un listado en forma de árbol de los directorios y archivos que acabas de crear.

**Paso 2)** Desde tu máquina física abre un navegador y establece las siguientes conexiones (donde 172.26.2.X es la dirección IP de tu máquina servidor Linux).

- <http://172.26.2.X>
- <http://172.26.2.X/despliegue.html>
- <http://172.26.2.X/ciclos>
- <http://172.26.2.X/ciclos/listado.html>



**Listado PROFE**



**Despliegue PROFE**



**Toma capturas de pantallas del listado en forma de árbol del paso 1 y de cada una de las conexiones del paso 2.**

## D.2) Archivos a servir por defecto (Directory Index).

Cuando hemos accedido directamente a la dirección IP de nuestra máquina Linux(172.26.2.X) sin pedir ningún recurso en concreto, el servidor ha enviado por defecto el archivo que ya existía en el directorio `/var/www/html`, o sea, el **index.html**. Esto es debido a que la directiva **DirectoryIndex** tiene asociado por defecto este archivo inicial.

Si el archivo **index.html** no existiera, el servidor por Apache serviría el listado de contenido de la carpeta `/var/www/html`.

**Paso 1)** Renombra el fichero **index.html** con el nombre **índice.html**

**Paso 2)** Ahora accede a la dirección IP de tu máquina Servidor Linux y observa que sirve el listado de recursos de la carpeta `/var/www/html`.

### Toma capturas de pantallas del paso 2.

Ahora vamos a cambiar el archivo que Apache sirve por defecto a `despliegue.html`:

**Paso 3)** Edita el archivo **000-default.conf** y añade la siguiente directiva **<Directory>** dentro de la directiva **<VirtualHost>**:

```
# 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>
    DirectoryIndex despliegue.html
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
</VirtualHost>
```

**Paso 4)** Reinicia el servidor Apache (si todo está correcto, no debe darte ningún problema el reiniciar el servidor, si falla el reinicio, seguramente es que te has equivocado escribiendo algo).

**`sudo service apache2 restart`**

**Paso 5)** Desde tu máquina física abre un navegador y establece una conexión con la IP de tu servidor Linux. Ahora el servidor deberá servirte el contenido de `despliegue.html`.

### Toma capturas de pantallas de los pasos 3 y 5.

## D.3) Opciones sobre directorios (Options Indexes).

Al cambiar las opciones para `/var/www/html`, éstas se heredan para todas las subcarpetas contenidas en ella (en nuestro caso, `/var/www/html/ciclos`), si accedemos a esta carpeta, como en `ciclos` no hay ningún archivo `despliegue.html` lo que hace es servir el contenido de la carpeta tal cual.

Si queremos que las opciones de las subcarpetas sean distintas debemos crear una directiva **<Directory>** para estas subcarpetas.

Si ponemos dentro de la directiva **Options** la palabra **Indexes** estamos permitiendo listar el contenido del directorio que hay dentro de la directiva. Si no ponemos esta opción, el acceso al recurso no estará permitido.

**Paso 1)** Desde la máquina física **accede al recurso ciclos** que hay en tu máquina servidor Linux. Como ha heredado la configuración debería servir el archivo



despliegue.html, pero como no existe sirve el contenido de la carpeta /var/www/ciclos/.

**Paso 2)** Vamos a crear una nueva directiva Directory para /var/www/ciclos/

```
#Include conf-available/serve-cgi-bin.conf

<Directory /var/www/html>
    DirectoryIndex despliegue.html
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>

<Directory /var/www/html/ciclos>
    Options FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
</VirtualHost>
```

**No hemos añadido Indexes en Options**, por lo que el acceso al recurso está prohibido.

**Paso 3)** Reinicia el servidor Apache.

**Paso 4)** Abre un navegador en tu máquina física y establece una conexión con el recurso ciclos que hay en tu máquina Linux. El acceso debe estar prohibido (forbidden).

**Toma capturas de pantallas de los pasos 2 y 4.**

**Paso 5)** Añade a la directiva de /var/www/html/ciclos la opción **DirectoryIndex** para que sirva el archivo **daw.html**.

**Paso 6)** Reinicia el servidor Apache.

**Paso 7)** Abre un navegador en tu máquina física y establece una conexión con el recurso ciclos que hay en tu máquina Linux. Ahora servirá la página daw.html.

**Toma capturas de pantallas de los pasos 5 y 7.**

#### D.4) Codigos de error (ErrorDocument).

La directiva *ErrorDocument* nos permite añadir un texto cuando se sirve un contenido que no está disponible en el servidor Apache. Por defecto, el mensaje que muestra será *NotFound* (error 404 en el protocolo HTTP). En este apartado vamos a modificar el mensaje de error.

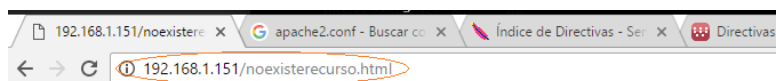
**Paso 1)** Edita el fichero **000-default.conf** y añade la directiva *ErrorDocument* junto con un mensaje de página no encontrada.

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

ErrorDocument 404 "<H1>Esta pagina no se puede servir porque no existe</H1>"
</VirtualHost>
```

**Paso 2)** Reinicia el servidor Apache para que los cambios surtan efecto.

**Paso 3)** Desde la máquina física, accede a la IP de tu máquina Linux solicitando el contenido de un recurso que no existe (por ejemplo noexisterecurso.html).



**Esta pagina no se puede servir porque no existe**

**Toma capturas de pantallas de los pasos 1 y 3.**

También se puede servir como contenido de un fichero:

**Paso 4)** Configura el servidor virtual por defecto para que cuando retorne el código de error 404 envíe la página 404.html almacenada en el directorio raíz del servidor.

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

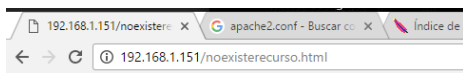
ErrorDocument 404 /404.html_

</VirtualHost>
```

**Paso 5)** Crea el fichero `/var/www/html/404.html` y añade dentro un mensaje de error.

**Paso 6)** Reinicia el servidor Apache para que los cambios tengan efecto.

**Paso 7)** Desde la máquina física, accede a la IP de tu máquina Linux solicitando el contenido de un recurso que no existe (por ejemplo `noexisterecurso.html`).



Este recurso no existe

[Prueba con otro recurso](#)

[Toma capturas de pantallas de los pasos 4,5 y 7.](#)

## D.5) Directorios virtuales (Alias).

La carpeta desde donde se sirve contenido es `/var/www/html`, pero podemos usar otros directorios desde los cuales servir contenido haciendo uso de los directorios virtuales. Para ello podemos añadir una directiva `<Directory>` y justo antes añadir la directiva `Alias` indicando desde qué directorio se va a servir el recurso indicado. A ojos del cliente HTTP, los recursos se servirán desde el mismo directorio, pero realmente se están sirviendo desde ubicaciones distintas.

**Paso 1)** Crea un directorio `/apuntes` en la carpeta home de tu usuario y dentro crea un fichero que se llame `apuntesTema1.html`, añade el contenido que quieras.

**Paso 2)** Edita el fichero `000-default.conf`. Usa una directiva Alias para crear un directorio virtual denominado `/apuntes` que referencie a tu carpeta `/home/tuUsuario/apuntes`. Usa la directiva `<Directory>` para definir las opciones de configuración del contenido a servir.

```
Require all granted
</Directory>

Alias /apuntes /home/profe/apuntes
<Directory /home/profe/apuntes>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>

ErrorDocument 404 /404.html

</VirtualHost>
```

**Paso 3)** Reinicia el servidor Apache para que los cambios surtan efecto.

**Paso 4)** Desde la máquina física, accede a la IP de tu máquina Linux solicitando el directorio `/apuntes`. El servidor Apache debe servir el contenido de la carpeta.

[Toma capturas de pantallas de los pasos 2 y 4.](#)

## D.6) Redirecciones (Redirect).

Con la directiva *Redirect* podemos redireccionar un recurso ficticio a cualquier página que haya en internet.

**Paso 1)** Edita el fichero **000-default.conf** y añade la siguiente línea en <VirtualHost>:

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

Redirect /periodico http://www.diariodesevilla.es_

ErrorDocument 404 /404.html
</VirtualHost>
```

**Paso 2)** Reinicia el servidor Apache para que los cambios tengan efecto.

**Paso 3)** Desde la máquina física, accede a la IP de tu máquina Linux solicitando el recurso **/periodico**. El servidor Apache servirá la página a la que hemos redireccionado el recurso pedido.



**Toma capturas de pantallas de los pasos 1 y 3.**

## E) Configuración básica de Apache en Windows2008xx

En esta parte de la práctica haremos algunos de los apartados del punto anterior pero aplicados en la instalación Apache de nuestro servidor Windows.

### E.1) Ficheros y directorios de prueba.

**Paso 1)** En nuestro Servidor Windows vamos a crear los siguientes directorios y archivos en la carpeta **C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\htdocs\**

Añade a los archivos contenido HTML (usa encabezados H1), el que quieras, siempre que no hiera la sensibilidad de quien lo lea (**¡cuidado que los archivos no sean .txt sino .html!**).

- ...htdocs/despliegue.html
- ...htdocs/fp.html
- ...htdocs/ciclos/listado.html
- ...htdocs/ciclos/asir.html
- ...htdocs/ciclos/daw.html
- ...htdocs/ciclos/dam.html

**Paso 2)** Crea en la carpeta **C:\Usuarios\Administrator** una carpeta **apuntes** y dentro un archivo **apuntes.html** con el contenido que quieras.

**Paso 3)** Desde tu máquina física abre un navegador y establece las siguientes conexiones (donde 172.26.2.X es la dirección IP de tu máquina servidor Windows).

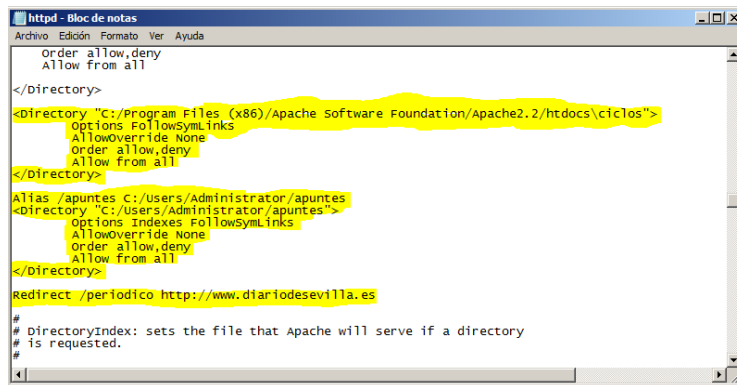
- <http://172.26.2.X>
- <http://172.26.2.X/despliegue.html>
- <http://172.26.2.X/ciclos>

- <http://172.26.2.X/ciclos/listado.html>

Toma capturas de pantallas de las carpetas y archivos creadas en el paso 1 y paso 2 y de cada una de las conexiones del paso 3.

## E.2) Configuración.

**Paso 1)** Edita el fichero **httpd.conf** y crea las directivas necesarias tal y como se muestran en las siguientes figuras (justo debajo de la directiva **<Directory "C:/.../htdocs">... </Directory>** que ya hay en el archivo):



Además, indica como el archivo por defecto a servir **despliegue.html** en la directiva **DirectoryIndex**.

```
#
# DirectoryIndex: sets the file that Apache will serve if a directory
# is requested.
#
<IfModule dir_module>
    DirectoryIndex despliegue.html
</IfModule>
```

**Paso 2)** Reinicia el servidor apache para que los cambios surtan efecto.



Si el servidor falla al reiniciarlo, probablemente te has equivocado al escribir alguna de las directivas.

En el archivo de error que está en:

**C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\logs**

aparecerán los posibles errores que has tenido al modificar el código de httpd.conf.

**Paso 3)** Desde tu máquina física abre un navegador y establece las siguientes conexiones (donde 172.26.2.X es la dirección IP de tu máquina servidor Windows).

- <http://172.26.2.X>(debe servir la página despliegue.html)
- <http://172.26.2.X/apuntes>(debe servir la página apuntes.html)
- <http://172.26.2.X/ciclos>(debe aparecer Forbidden al no incluir Indexes)
- <http://172.26.2.X/periodico>(debe servir la página www.diariodesevilla.es)

Toma una captura del paso 1 y de cada una de las conexiones del paso 3.

## F) Módulos en Linux

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

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

**sudo apache 2ctl -l**

También existen módulos dinámicos, los cuales pueden cargarse y descargarse de manera dinámica. En Linux, los módulos disponibles se encuentran en el directorio **/etc/apache2/mods-available/**

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

Mientras que los módulos que están cargados se encuentran en el directorio **/etc/apache2/mods-enabled/**

Para habilitar y deshabilitar módulos se usan los comandos:

**a2enmod nombre\_del\_modulo**

**a2dismod nombre\_del\_modulo**

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

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

### F.1) Módulos

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

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

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

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

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

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

### F.2) Módulo userdir

El módulo **userdir** se usa 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 el raíz 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?

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

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


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

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

**PASO 6)** Crea el directorio necesario dentro de tu usuario y añade un fichero denominado **personal.html** con el contenido que quieras

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

### Index of /~profe

Name	Last modified	Size	Description
<hr/>			
 <a href="#">Parent Directory</a>	-		
 <a href="#">personal.html</a>	2016-11-22 19:55	38	
<hr/>			
Apache/2.4.18 (Ubuntu) Server at 192.168.1.151 Port 80			

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

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

### G) Módulos en Windows2008xx

En Windows, los módulos estáticos se pueden consultar desde una consola de comandos mediante el comando **httpd -l**.

Los módulos dinámicos están en la siguiente ubicación:

**C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\modules\**

Estos módulos se pueden cargar descomentando el módulo que queremos usar (quitando el carácter de comentario #) y se pueden descargar comentando el módulo que queremos usar (añadiendo el carácter de comentario #) en el archivo:

**C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\httpd.conf**

#### G.1) Módulos

**PASO 1)** En el servidor Windows 2008 abre una consola de comandos y comprueba los módulos estáticos que se han cargado al compilar el servidor Apache.

**PASO 2)** Consulta el fichero desde donde se cargan los módulos dinámicos. ¿Qué módulos se encuentran cargados?

**PASO 3)** Consulta el directorio donde se encuentran los módulos disponibles para cargar.

**Toma una captura de los pasos 1, 2 y 3.**

## G.2) Módulo userdir

**PASO 1)** Habilita el módulo userdir descomentando las directivas **LoadModule** (para mod\_userdir.so) e **Include** (para /conf/extra/httpd-userdir.conf) en el archivo correspondiente.

**PASO 2)** Abre el fichero **httpd-userdir.conf** (puedes ver la ubicación del archivo en el paso anterior). Observa que está habilitado el uso de directorios personales para todos los usuarios. ¿Qué subdirectorio deben crear los usuarios para poner sus páginas personales?

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

Si el servidor falla al reiniciarlo, probablemente te has equivocado al escribir alguna de las directivas.

En el archivo de error que está en:

**C:\Program Files(x86)\Apache Software Foundation\Apache 2.2\logs**

aparecerán los posibles errores que has tenido al modificar el código de httpd.conf.

**PASO 4)** Crea, para tu usuario en el directorio correspondiente, el subdirectorio para poder añadir tus páginas personales. Crea dentro el archivo **personal.html** con el contenido que quieras.

**PASO 5)** Desde la máquina física, abre un navegador y accede al directorio raíz de tu usuario Windows.

**PASO 6)** Vuelve a descargar el módulo userdir y reinicia el servidor para que los cambios tengan efecto.

**Toma una captura de los pasos 1, 2, 3 y 5 7 (en esta última, donde se vea la barra de direcciones del navegador).**

- *Crea un fichero que se llame Practica3Apellido1Apellido2.pdf .*  
*Inserta todas las capturas de pantallas por orden explicando cada una de ellas.*
- *Una vez terminada la práctica, sube el archivo a la classroom.*