

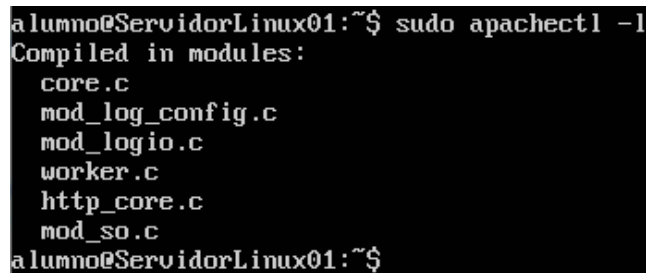
Práctica 5.7: Módulos en *Linux*

Consulta los módulos estáticos y los módulos dinámicos cargados por defecto en la versión de *Apache* instalada en la máquina **ServidorLinuxXX**. Posteriormente, investiga la funcionalidad del módulo *userdir*, habilitalo y prueba su funcionalidad.

1. Módulos

- 1.1. Inicia una sesión en **ServidorLinuxXX** con un usuario con privilegios de administrador.
- 1.2. Comprueba los módulos estáticos que se han cargado al compilar el servidor ejecutando el siguiente comando, Figura 1:


```
sudo apache2ctl -l
```



```
alumno@ServidorLinux01:~$ sudo apache2ctl -l
Compiled in modules:
  core.c
  mod_log_config.c
  mod_logio.c
  worker.c
  http_core.c
  mod_so.c
alumno@ServidorLinux01:~$
```

Figura 1: Módulos estáticos

- 1.3. Comprueba los módulos que se han cargado dinámicamente al arrancar el servidor consultando el directorio **/etc/apache2/mods-enabled**. Observa que los ficheros que aparecen en este directorio son enlaces simbólicos a ficheros de **/etc/apache2/mods-available**.
- 1.4. Edita uno de los ficheros **.load** (por ejemplo **dir.load**) y observa cómo se utiliza la directiva **LoadModule**, Figura 2 para cargar el módulo. Comprueba cuál es la ruta donde está el código del módulo (archivo **.so**).



```
LoadModule dir_module /usr/lib/apache2/modules/mod_dir.so
```

Figura 2: Fichero **dir.load**

- 1.5. Edita uno de los ficheros **.conf** (por ejemplo **dir.conf**) y observa cómo se añaden directivas dentro de una declaración **<IfModule nombremodulo> ... </IfModule>**}, Figura 3 que se ejecutarán si se carga el módulo.

```
<IfModule mod_dir.c>
    DirectoryIndex index.html index.cgi index.pl index.php index.xhtml
</IfModule>
```

Figura 3: Fichero **dir.conf**

- 1.6. Consulta el directorio `/usr/lib/apache2/modules/` y observa los módulos disponibles para cargar.
- 1.7. Ejecuta el siguiente comando para mostrar los paquetes disponibles en los repositorios de *Ubuntu* que permiten instalar módulos adicionales en *Apache*, Figura 4.

```
sudo apt-cache search libapache2-mod
```

```
libapache2-mod-apparmor - changehat AppArmor library as an Apache module
libapache2-mod-perl2-doc - Integration of perl with the Apache2 web server -
documentation
libapache2-mod-php5 - server-side, HTML-embedded scripting language (Apache 2
module)
libapache2-mod-wsgi - Python WSGI adapter module for Apache
php5-cgi - server-side, HTML-embedded scripting language (CGI binary)
libapache2-mod-auth-kerb - modulo apache2 para la autenticación en Kerberos
libapache2-mod-auth-mysql - módulo apache 2 para autenticación MySQL
libapache2-mod-auth-pgsql - Módulo para Apache2 que proporciona autenticación
sql
libapache2-mod-auth-plain - Módulo para Apache2 que provee de autenticación e
texto plano
libapache2-mod-axis2c - Motor de servicios web Apache - módulo de apache
libapache2-mod-macro - Use macros en los archivos de configuración de apache2
libapache2-mod-perl2 - Integración de perl con el servidor web Apache 2
libapache2-mod-perl2-dev - Integración de perl con el servidor web Apache 2 -
chivos de desarrollo
libapache2-mod-python - módulo integrado Python para Apache 2
libapache2-mod-python-doc - Modulo de empujado Python para Apache 2 - docume
ción
libapache2-mod-fastcgi - módulo FastCGI de Apache 2 para scripts CGI de gran
cucion
libapache2-mod-apreq2 - generic Apache request library - Apache module
libapache2-mod-auth-cas - CAS authentication module for Apache2
```

Figura 4: Módulos disponibles en los repositorios de *Ubuntu*

2. Módulo *userdir*

- 2.1. Comprueba, consultando el directorio `/etc/apache2/mods-enabled` que el módulo *userdir* no está habilitado.
- 2.2. Habilita el módulo ejecutando el siguiente comando:

```
sudo a2enmod userdir
```

- 2.3. Verifica que dentro del directorio `/etc/apache2/mods-enabled` se han creado enlaces simbólicos del módulo *userdir* (archivos `.conf` y `.load`) hacia `/etc/apache2/mod_`*available*s, Figura 5.

```
lrwxrwxrwx 1 root root 30 may  4 23:09 userdir.conf -> ../mods-available/userdir.conf
lrwxrwxrwx 1 root root 30 may  4 23:09 userdir.load -> ../mods-available/userdir.load
alumno@ServidorLinux01:/etc/apache2/mods-enabled$
```

Figura 5: Módulo **userdir** habilitado

- 2.4. Reinicia el servidor para que los cambios tengan efecto.
- 2.5. Consulta el fichero `/etc/apache2/mod.enabled/userdir.conf`. Observa que está habilitado el uso de directorios personales para todos los usuarios excepto para el usuario **root** y que **public_html** es el nombre del subdirectorio que pueden crear los usuarios en su directorio **home** para poner sus páginas personales.
- 2.6. Crea en directorio `/home/alumno/public_html`. Dentro del directorio crea un fichero denominado **personal.html** con el contenido que quieras.
- 2.7. Desde **DesarrolloW7XX** accede a `http://192.168.1.X7/~alumno`, Figura 6.

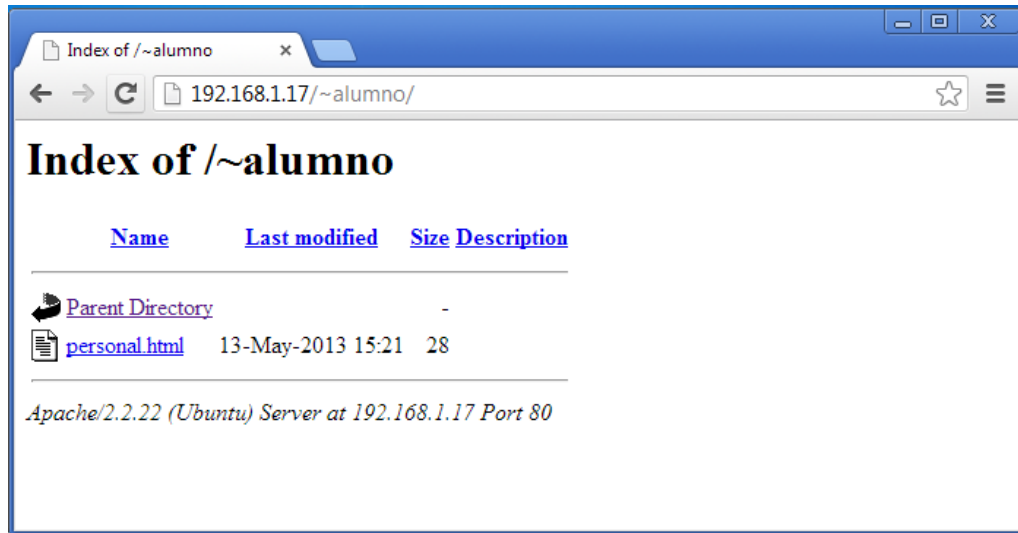


Figura 6: Acceso al directorio personal del usuario alumno

- 2.8. Deshabilita el módulo ejecutando el siguiente comando:

```
sudo a2dismod userdir
```

- 2.9. Reinicia el servidor para que los cambios tengan efecto.

◇