

Módulos

Un Módulo en apache es una forma de agrupar y de modularizar determinadas funciones del servidor. El objeto que tiene es poder personalizar cada instalación de Apache de forma que podamos cargar las funciones de éste que nos hagan falta.

- Los estáticos son los que se cargan durante la instalación de Apache se pueden comprobar con el siguiente comando: **sudo apachectl -l**

```
profesor@ubuntu:~$ sudo apachectl -l
[sudo] password for profesor:
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
profesor@ubuntu:~$
```

Arriba aparecen los módulos estáticos que se cargan durante la instalación de Apache.

- Los dinámicos son los que se pueden cargar posteriormente a la instalación o sea en cualquier momento . para saber los módulos de que disponemos podemos ir a la carpeta /etc/apache2/mods-available/

En la imagen aparecen algunos de los módulos disponibles para apache

```
profesor@ubuntu:~$ ls /etc/apache2/mods-available/
access_compat.load  cgid.load  ldap.load  ratelimit.load
actions.conf        cgi.load   log_debug.load  reflector.load
actions.load        charset_lite.load  log_forensic.load  remoteip.load
alias.conf          data.load  lua.load      reqtimeout.conf
alias.load          dav_fs.conf  macro.load    reqtimeout.load
allowmethods.load  dav_fs.load  mime.conf     request.load
asis.load           dav.load    mime.load     rewrite.load
auth_basic.load     dav_lock.load  mime_magic.conf  sed.load
```

En el directorio /etc/apache2/mods-enabled se encuentran los módulos que se han cargado dinámicamente.

```
profesor@ubuntu:~$ ls /etc/apache2/mods-enabled/
access_compat.load  authn_file.load  autoindex.load
alias.conf          authz_core.load  deflate.conf
alias.load          authz_host.load  deflate.load
auth_basic.load     authz_user.load  dir.conf
authn_core.load     autoindex.conf  dir.load
```

Puedes observar que cada módulo tiene su par de ficheros con extensión .load y .conf
.load → Se encarga de cargar el módulo. Observa su interior

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

.conf → Se encarga de enlazar la directiva al fichero de configuración apache2.conf

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

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

Módulos en Apache con Linux

Podemos consultar el directorio: `/usr/lib/apache2/modules`

Y podrás ver una lista de módulos disponibles para cargar

También puedes buscar los paquetes disponibles en los repositorios de Ubuntu que permiten instalar módulos adicionales en Apache.

Sudo apt-cache search libapache2-mod

```
libapache2-mod-auth-plain - Módulo para Apache2 que provee de autenticación en texto plano
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 - archivos de desarrollo
libapache2-mod-perl2-doc - integración de perl con el servidor web Apache 2 - documentación
libapache2-mod-php7.0 - lado servidor, lenguaje de guionización HTML-embuido (módulo Apache 2)
libapache2-mod-python - módulo integrado Python para Apache 2
libapache2-mod-python-doc - Módulo de empotrado Python para Apache 2 - documentación
libapache2-mod-wsgi - Módulo adaptador Python WSGI para Apache
libapache2-mod-apparmor - changehat AppArmor library as an Apache module
```

Aparecen algunos en la imagen superior junto a su descripción

Ejemplo de uso de un módulo:

1. Userdir es módulo que sirve para hacer que cada usuario (salvo root) que tenga una cuenta en el sistema podrá tener su rincón en el servidor mediante al que se accederá poniendo el nombre en la barra de direcciones.
2. Comprueba que el módulo userdir no está habilitado, para ello deberemos comprobar que no está en el directorio `/etc/apache2/mods-enabled`

```
profesor@ubuntu:~$ ls /etc/apache2/mods-enabled/
access_compat.load  authn_file.load  autoindex.load  env.load  mpm_event.load  status.conf
alias.conf          authz_core.load  deflate.conf    filter.load  negotiation.conf status.load
alias.load          authz_host.load  deflate.load     mime.conf  negotiation.load
auth_basic.load     authz_user.load  dir.conf        mime.load   setenvif.conf
authn_core.load     autoindex.conf  dir.load        mpm_event.conf setenvif.load
```

3. Al no estar lo habilitamos con el comando: **sudo a2enmod userdir**

```
profesor@ubuntu:~$ sudo a2enmod userdir
Enabling module userdir.
To activate the new configuration, you need to run:
service apache2 restart
```

4. Verificamos que se han creado enlaces simbólicos del módulo userdir (ficheros `.conf` y `.load`) hacia `/etc/apache2/mod-available` en el otro directorio.

```
lrwxrwxrwx 1 root root 30 oct 18 16:04 userdir.conf -> ../mods-available/userdir.conf
lrwxrwxrwx 1 root root 30 oct 18 16:04 userdir.load -> ../mods-available/userdir.load
```

5. Reiniciamos el servidor.

6. Visualizamos el fichero de configuración del módulo recientemente cargado.

```
<IfModule mod_userdir.c>
    UserDir public_html
    UserDir disabled root

    <Directory /home/*/public_html>
        AllowOverride FileInfo AuthConfig Limit Indexes
        Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
        <Limit GET POST OPTIONS>
            Require all granted
        </Limit>
        <LimitExcept GET POST OPTIONS>
            Require all denied
        </LimitExcept>
    </Directory>
</IfModule>

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

Observa que el único usuario que no puede usar el módulo es root.

El directorio más importante es **public_html** es el directorio que contendrá las páginas a mostrar del usuario y debe estar en la carpeta personal del usuario.

Las demás indicaciones son referentes a los permisos de la carpeta **public_html**