

<b>Índice</b>	<b>2</b>
<b>1. Introducción</b>	<b>3</b>
<b>2. Instalación</b>	<b>3</b>
2.1. Estructura del directorio /etc/apache2 . . . . .	3
<b>3. Módulos de apache</b>	<b>4</b>
<b>4. Utilización de módulos</b>	<b>6</b>

# Utilización de módulos en Apache

# 1. Introducción

Uno de los aspectos característicos del servidor HTTP Apache (en adelante sólo Apache) es su modularidad, Apache tiene un sinfín de características adicionales que si estuvieran siempre incluidas, harían de él un programa demasiado grande y pesado. En lugar de esto, Apache se compila de forma modular y se cargan en memoria sólo los módulos necesarios en cada caso.

## 2. Instalación

La instalación básica de apache se realiza con la siguiente instrucción<sup>1</sup>:

```
avatar:~# aptitude install apache2
```

```
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Leyendo la información de estado extendido
Inicializando el estado de los paquetes... Hecho
Leyendo las descripciones de las tareas... Hecho
Se instalarán los siguiente paquetes NUEVOS:
  apache2 apache2-mpm-worker{a} apache2-utils{a} apache2.2-common{a}
  libapr1{a} libaprutil1{a} libexpat1{a} libmysqlclient15off{a}
  libpq5{a} mysql-common{a} openssl{a} openssl-blacklist{a} psmisc{a}
  ssl-cert{a}
0 paquetes actualizados, 14 nuevos instalados, 0 para eliminar y 0 \
sin actualizar.
Necesito descargar 11,5MB de ficheros. Después de desempaquetar se \
usarán 26,6MB.
¿Quiere continuar? [Y/n/?]
```

Una vez instalado el paquete apache y todas sus dependencias, tendremos el servidor web Apache totalmente operativo en avatar y accesible a través de las URL <http://localhost> o <http://avatar.example.com>.

### 2.1. Estructura del directorio /etc/apache2

En versiones anteriores de Apache había un solo fichero de configuración (/etc/httpd.conf o /etc/apache.conf), que fue creciendo poco a poco y era cada vez más incómodo de manejar. En la versión 2 de Apache y en particular en el empaquetamiento que se ha hecho en Debian, este fichero único se ha sustituido por un directorio con varios ficheros de configuración:

**/etc/apache2/apache2.conf** Fichero principal de configuración y en el que aparecen el resto de ficheros de configuración mediante directivas Include.

**/etc/apache2/httpd.conf** Fichero inicialmente vacío y que se incluye en Debian por compatibilidad con versiones anteriores.

**/etc/apache2/ports.conf** Fichero que define los puertos por los que va a aceptar peticiones Apache (inicialmente sólo 80/tcp).

<sup>1</sup>Aunque hasta hace poco se ha mantenido el desarrollo de la rama 1 de Apache, para servidores web nuevos es siempre recomendable utilizar la rama 2

**/etc/apache2/mods-available/** Directorio que contiene los módulos disponibles en la instalación actual.

**/etc/apache2/mods-enabled/** Directorio que incluye mediante enlaces simbólicos al directorio anterior, los módulos que se van a cargar en memoria la próxima vez que se inicie Apache.

**/etc/apache2/sites-available/** Directorio que contiene los sitios disponibles en la instalación actual. La instalación básica de Apache en Debian incluye dos sitios:

**default** Sitio por defecto para el puerto 80/tcp

**default-ssl** Sitio por defecto para el puerto 443/tcp

**/etc/apache2/sites-enabled/** Directorio que incluye mediante enlaces simbólicos al directorio anterior, los sitios que va a utilizar Apache la próxima vez que se inicie. Inicialmente el único sitio web activo es 000-default que es un enlace simbólico a default

### 3. Módulos de apache

Los módulos de apache se pueden encontrar de dos maneras, compilados dentro del ejecutable apache2 o compilados de forma individual como una biblioteca de enlace dinámico (con extensión .so). Para saber qué módulos incluye el ejecutable de nuestra instalación de apache, podemos utilizar la siguiente instrucción:

```
avatar:~# apache2 -l
```

```
Compiled in modules:
  core.c
  mod_log_config.c
  mod_logio.c
  worker.c
  http_core.c
  mod_so.c
```

El resto de módulos disponibles para cargar en tiempo de ejecución se encuentran en el directorio:

```
avatar:~# ls /usr/lib/apache2/modules/
```

```
httpd.exp          mod_cgi.so        mod_mime_magic.so
mod_actions.so    mod_charset_lite.so  mod_mime.so
mod_alias.so      mod_dav_fs.so     mod_negotiation.so
mod_asis.so       mod_dav_lock.so   mod_proxy_ajp.so
mod_auth_basic.so mod_dav.so        mod_proxy_balancer.so
mod_auth_digest.so mod_dbd.so        mod_proxy_connect.so
mod_authn_alias.so mod_deflate.so    mod_proxy_ftp.so
mod_authn_anon.so  mod_dir.so       mod_proxy_http.so
mod_authn_dbd.so   mod_disk_cache.so mod_proxy.so
mod_authn_dbm.so   mod_dumpio.so    mod_rewrite.so
mod_authn_default.so mod_env.so       mod_setenvif.so
mod_authn_file.so  mod_expires.so   mod_speling.so
mod_authnz_ldap.so mod_ext_filter.so mod_ssl.so
mod_authz_dbm.so   mod_file_cache.so mod_status.so
mod_authz_default.so mod_filter.so    mod_substitute.so
mod_authz_groupfile.so mod_headers.so  mod_suexec.so
mod_authz_host.so  mod_ident.so     mod_unique_id.so
```

```

mod_authz_owner.so      mod_imagemap.so      mod_userdir.so
mod_authz_user.so      mod_include.so       mod_usertrack.so
mod_autoindex.so      mod_info.so          mod_version.so
mod_cache.so           mod_ldap.so          mod_vhost_alias.so
mod_cern_meta.so       mod_log_forensic.so
mod_cgid.so            mod_mem_cache.so

```

Pueden parecer muchos, pero son sólo los módulos de la instalación estándar y se incluyen dentro del paquete apache2.2-common. Hay otros muchos módulos que se distribuyen en paquetes separados, que en debian reciben el nombre libapache2-mod-\*:

```
avatar:~# aptitude search libapache2-mod
```

```

p  libapache2-authcassimple-perl
p  libapache2-authenntlm-perl
p  libapache2-mod-apreq2
p  libapache2-mod-auth-kerb
p  libapache2-mod-auth-mysql
p  libapache2-mod-auth-openid
p  libapache2-mod-auth-pam
p  libapache2-mod-auth-pgsql
p  libapache2-mod-auth-plain
p  libapache2-mod-auth-radius
p  libapache2-mod-auth-sys-group
p  libapache2-mod-bw
p  libapache2-mod-chroot
p  libapache2-mod-defensible
p  libapache2-mod-dnssd
p  libapache2-mod-encoding
p  libapache2-mod-evasive
p  libapache2-mod-fastcgi
p  libapache2-mod-fcgid
p  libapache2-mod-geoip
p  libapache2-mod-gnutls
p  libapache2-mod-jk
p  libapache2-mod-layout
p  libapache2-mod-ldap-userdir
p  libapache2-mod-line-edit
p  libapache2-mod-lisp
p  libapache2-mod-log-sql
p  libapache2-mod-log-sql-dbi
p  libapache2-mod-log-sql-mysql
p  libapache2-mod-log-sql-ssl
p  libapache2-mod-macro
p  libapache2-mod-mime-xattr
p  libapache2-mod-mono
p  libapache2-mod-musicindex
p  libapache2-mod-neko
p  libapache2-mod-ocamlnet
p  libapache2-mod-perl2
p  libapache2-mod-perl2-dev
p  libapache2-mod-perl2-doc
p  libapache2-mod-php5
p  libapache2-mod-php5filter
p  libapache2-mod-proxy-html
p  libapache2-mod-python
p  libapache2-mod-python-doc

```

```

v  libapache2-mod-python2.5
p  libapache2-mod-random
p  libapache2-mod-removeip
p  libapache2-mod-rpaf
p  libapache2-mod-ruby
p  libapache2-mod-scgi
p  libapache2-mod-shib
p  libapache2-mod-shib2
p  libapache2-mod-speedycgi
p  libapache2-mod-suphp
p  libapache2-mod-vhost-hash-alias
p  libapache2-mod-vhost-ldap
p  libapache2-mod-wsgi
p  libapache2-redirector
p  libapache2-reload-perl
p  libapache2-request-perl
p  libapache2-svn
p  libapache2-webauth
p  libapache2-webkdc

```

Como puede verse, las posibilidades que ofrece el servidor web Apache son realmente enormes y este es uno de los motivos por los que Apache es el servidor web más utilizado hoy día en la web <sup>2</sup>.

## 4. Utilización de módulos

Si vamos al directorio donde se ubican los módulos disponibles de Apache y hacemos un listado, obtenemos lo siguiente:

```
avatar:~# ls /etc/apache2/mods-available/
```

```

actions.conf      dav_fs.conf      mime.load
actions.load      dav_fs.load      mime_magic.conf
alias.conf        dav.load         mime_magic.load
alias.load        dav_lock.load    negotiation.conf
asis.load         dbd.load         negotiation.load
auth_basic.load   deflate.conf     proxy_ajp.load
auth_digest.load deflate.load      proxy_balancer.load
authn_alias.load dir.conf         proxy.conf
authn_anon.load  dir.load        proxy_connect.load
authn_dbd.load    disk_cache.conf proxy_ftp.load
authn_dbm.load    disk_cache.load proxy_http.load
authn_default.load dump_io.load     proxy.load
authn_file.load   env.load        rewrite.load
authnz_ldap.load  expires.load    setenvif.conf
authz_dbm.load    ext_filter.load setenvif.load
authz_default.load file_cache.load spelling.load
authz_groupfile.load filter.load      ssl.conf
authz_host.load   headers.load    ssl.load
authz_owner.load  ident.load      status.conf
authz_user.load   imagemap.load   status.load
autoindex.conf    include.load    substitute.load
autoindex.load    info.conf       suexec.load
cache.load        info.load       unique_id.load

```

<sup>2</sup><http://news.netcraft.com/archives/category/web-server-survey/>

```

cern_meta.load      ldap.load           userdir.conf
cgid.conf           log_forensic.load  userdir.load
cgid.load           mem_cache.conf     usertrack.load
cgi.load            mem_cache.load     version.load
charset_lite.load   mime.conf          vhost_alias.load

```

Los ficheros con extensión load suelen incluir una línea con la directiva LoadModule:

```

/etc/apache2/mods-available/userdir.load
1 LoadModule userdir_module /usr/lib/apache2/modules/mod_userdir.so

```

Además de cargar el módulo, en muchos casos es necesario realizar alguna configuración mediante directivas, por lo que en esos casos se existe un fichero con extensión .conf, como en el caso de userdir:

```

/etc/apache2/mods-available/userdir.conf
1 <IfModule mod_userdir.c>
2     UserDir public_html
3     UserDir disabled root
4
5     <Directory /home/*/public_html>
6         AllowOverride FileInfo AuthConfig Limit Indexes
7         Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
8         <Limit GET POST OPTIONS>
9             Order allow,deny
10            Allow from all
11        </Limit>
12        <LimitExcept GET POST OPTIONS>
13            Order deny,allow
14            Deny from all
15        </LimitExcept>
16    </Directory>
17 </IfModule>

```

Si queremos que Apache utilice cualquier módulo, lo que tendríamos que hacer es un enlace simbólico del fichero de extensión .load (y del .conf si existe), aunque podemos utilizar la instrucción a2enmod que facilita esta labor, por ejemplo para activar el módulo ldap en apache:

```
avatar:~# a2enmod
```

```

Your choices are: actions alias asis auth_basic auth_digest
authn_alias authn_anon authn_dbd authn_dbm authn_default authn_file
authnz_ldap authz_dbm authz_default authz_groupfile authz_host
authz_owner authz_user autoindex cache cern_meta cgi cgid charset_lite
dav dav_fs dav_lock dbd deflate dir disk_cache dump_io env expires
ext_filter file_cache filter headers ident imagemap include info
ldap log_forensic mem_cache mime mime_magic negotiation proxy
proxy_ajp proxy_balancer proxy_connect proxy_ftp proxy_http rewrite
setenvif speling ssl status substitute suexec unique_id userdir
usertrack version vhost_alias
Which module(s) do you want to enable (wildcards ok)?
ldap
Enabling module ldap.
Run '/etc/init.d/apache2 restart' to activate new configuration!

```

También disponemos de las instrucciones a2dismod para deshabilitar módulos, a2ensite para habilitar un sitio web y a2dissite para deshabilitarlo.