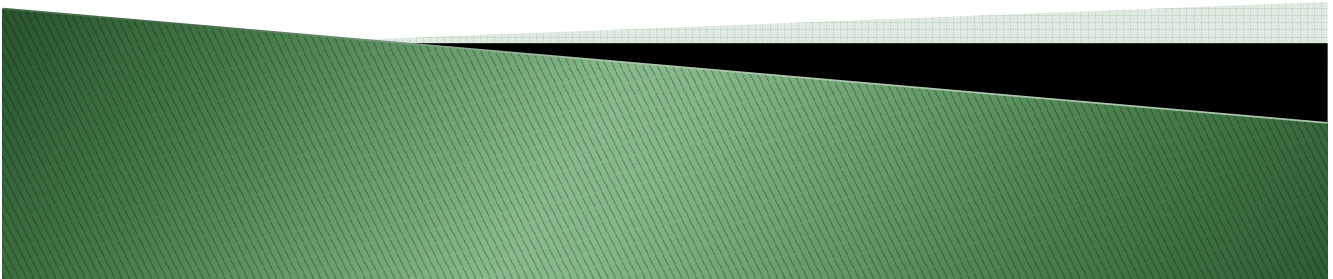


# Unidad 5

## Administración de *Apache* – 1

Despliegue de aplicaciones web



## Índice

---

- ▶ Introducción.
- ▶ Características.
- ▶ Versiones.
- ▶ Instalación.
- ▶ Inicio/parada del servidor
  - Introducción.
  - *Linux (Debian/Ubuntu).*
  - *Windows.*
- ▶ Servidor “principal” vs Servidores virtuales

# Índice

---

- ▶ Ficheros de configuración
  - Introducción.
  - *Linux (Debian/Ubuntu).*
  - *Windows.*
- ▶ Directivas.
- ▶ Configuración básica
- ▶ Módulos.
  - Introducción.
  - *Linux (Debian/Ubuntu).*
  - *Windows.*

# Índice

---

- ▶ Control del acceso.
- ▶ Autenticación y autorización.
- ▶ Ficheros .htaccess.
- ▶ Monitorización.
  - Ficheros de registros (logs).
  - Módulos para monitorizar el servidor.
  - Analizadores de *logs*.
- ▶ Bibliografía.

# Introducción

## ► Apache

- <http://www.apache.org>.
- Fundación que proporciona soporte a la comunidad con un numeroso conjunto de proyectos “*open source*”.
- Desarrolladores y usuarios caracterizados por ser un grupo colaborativo, con el deseo de crear software de alta calidad.
- <http://www.apache.org/foundation/>




The Apache Software Foundation

*Community-led development since 1999.*

# Introducción

## ► Proyectos

- <http://projects.apache.org/>

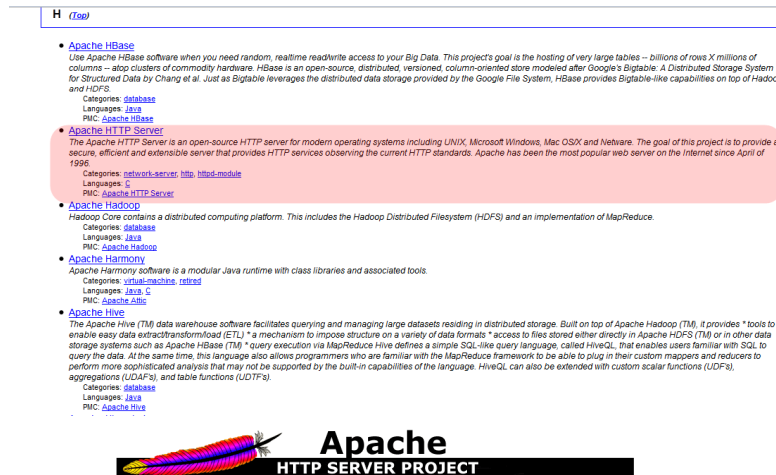


The screenshot shows the Apache Software Foundation website. At the top, there is a logo with a feather and the text "The Apache Software Foundation" and "http://www.apache.org/". Below the logo, there is a navigation menu with links: Home, Indexes, Feeds, DOAP Files, Documentation, and References & Thanks. The main content area is titled "Alphabetical Index" and contains a list of projects indexed on the site. The projects listed are: Anakia, Apache .NET Ant Library, Apache Abdera, Apache Accumulo, Apache ActiveMQ, and Apache Ant. Each project entry includes a brief description and a link to the project's page.

# Introducción

## ► HTTP Server

- <http://httpd.apache.org>.
- Uno de los principales proyectos



Despliegue de aplicaciones web

7

# Características

- Servidor Web potente, flexible y ajustado al HTTP/1.1.



- Modular.
  - Altamente configurable y extensible.
  - Ofrece su propio API (*Application Programming Interface*).
- Servidor “*Open Source*”.
- Multiplataforma: *Windows, Linux, Unix, ...*

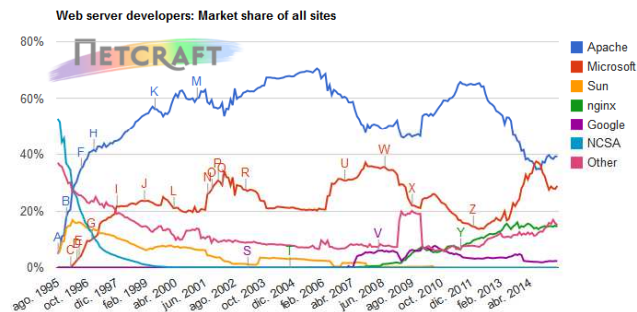
Despliegue de aplicaciones web

8

# Características

## ► Servidor “mas utilizado” en Internet (1)

- <http://www.netcraft.com>



Developer	April 2015	Percent	May 2015	Percent	Change
Apache	333,285,741	39.25%	336,813,959	39.26%	0.00
Microsoft	236,288,843	27.83%	247,784,668	28.88%	1.05
nginx	126,274,778	14.87%	123,697,645	14.42%	-0.45
Google	20,051,433	2.36%	20,103,068	2.34%	-0.02

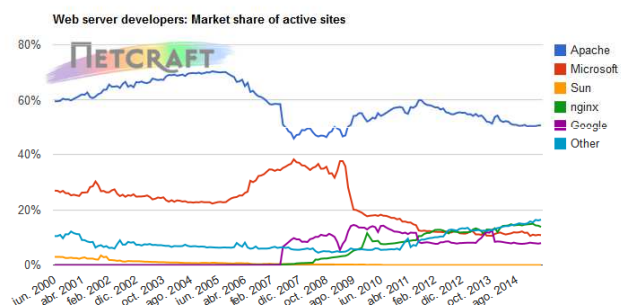
Despliegue de aplicaciones web

9

# Características

## ► Servidor “mas utilizado” en Internet (2)

- <http://www.netcraft.com>



Developer	April 2015	Percent	May 2015	Percent	Change
Apache	90,001,047	50.91%	89,387,114	50.98%	0.07
nginx	25,174,837	14.24%	24,204,572	13.81%	-0.43
Microsoft	19,353,327	10.95%	18,984,654	10.83%	-0.12
Google	13,712,694	7.76%	13,931,238	7.95%	0.19

Despliegue de aplicaciones web

10

# Versiones

---

- ▶ Versiones soportadas:
  - Versión 2.4.
  - Versión 2.2.
  - Versión 2.0.
  - Obsoleta
    - Versión 1.3.
- ▶ *Trunk* o versión en desarrollo.

- [Version 2.4](#)
- [Version 2.2](#)
- [Version 2.0](#)
- [Trunk](#) (development version)

# Instalación

---

- ▶ Tipos de instalación
  - Compilar código fuente.
  - Utilizar paquetes binarios adaptados para cada Sistema operativo (usaremos este tipo).

Stable Release - Latest Version:

- [2.4.12](#) (released 2015-01-29)

Legacy Release - 2.2 Branch:

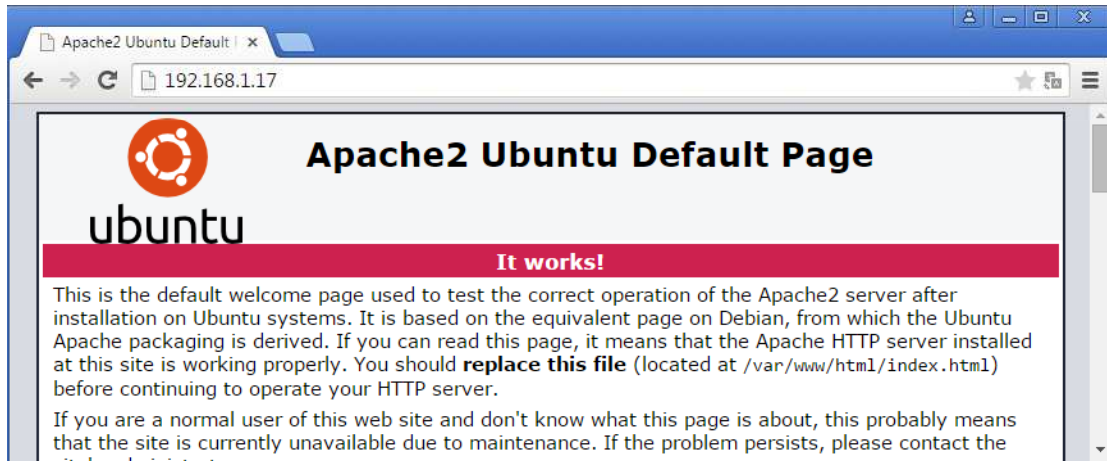
- [2.2.29](#) (released 2014-09-03)

If you are downloading the Win32 distribution, please read these [important notes](#).

# Práctica

## ► Práctica 5.1

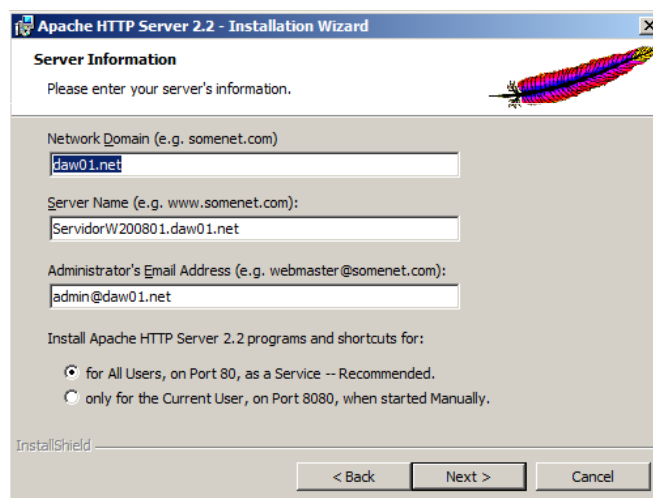
- Instalación y del servidor web *Apache 2.4* en *Linux*.



# Práctica

## ► Práctica 5.2

- Instalación y del servidor web *Apache 2.2* en *Windows*.



# Inicio/parada del servidor

## Introducción

---

- ▶ Cada que vez que cambien los ficheros de configuración.
- ▶ Hay que:
  - Parar/iniciar el servidor.
  - O indicarle que lea de nuevo los ficheros de configuración .
- ▶ Para que se apliquen los cambios.

# Inicio/parada del servidor

## *Linux (Debian/Ubuntu)*

---

- ▶ Inicio (varias posibilidades)
  - `sudo service apache2 start`
  - `sudo /etc/init.d/ apache2 start`
  - `sudo apachectl start`
- ▶ Parada (varias posibilidades)
  - `sudo service apache2 stop`
  - `sudo /etc/init.d/ apache2 stop`
  - `sudo apachectl stop`



# Inicio/parada del servidor

## *Linux (Debian/Ubuntu)*

---

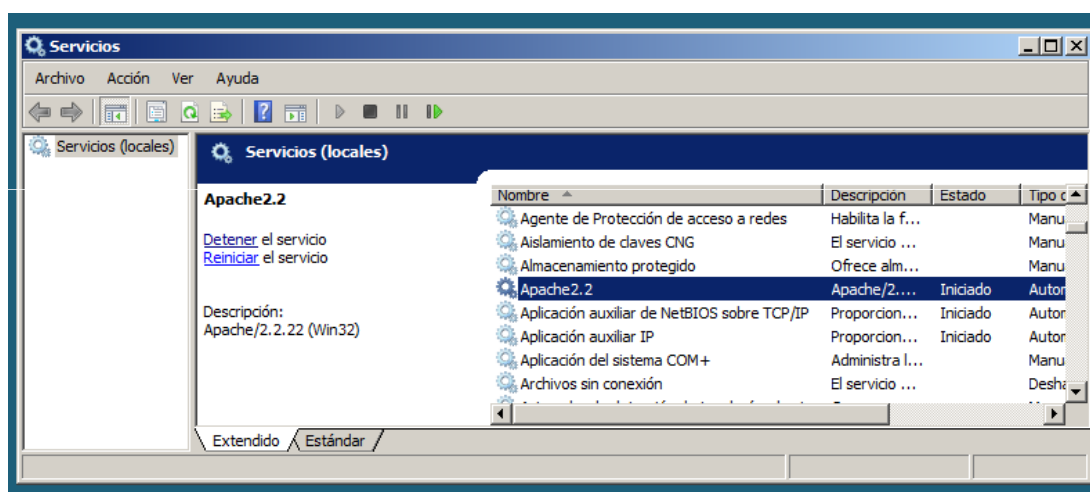
- ▶ Lea de nuevo los ficheros (varias posibilidades)
  - `sudo service apache2 reload`
  - `sudo /etc/init.d/ apache2 reload`

# Inicio/parada del servidor

## *Windows*

---

- ▶ Iniciar/parar el servicio.

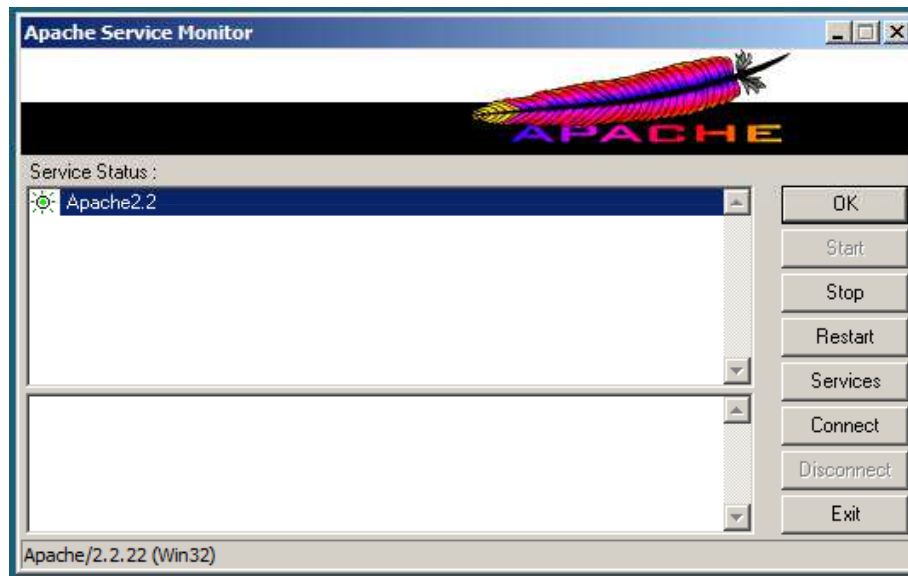


# Inicio/parada del servidor

## Windows

---

### ► *Apache Monitor*



# Inicio/parada del servidor

## Windows

---

- **Línea de comandos**
  - Inicio (varias posibilidades)
    - `net start apache2.2`
    - `httpd -k start`
  - Parada (varias posibilidades)
    - `net stop apache2.2`
    - `httpd -k stop`
    - `httpd -k shutdown`

# Servidor “principal” vs Servidores virtuales

---

## ▶ Servidor principal

- Atiende las peticiones si no se configuran servidores virtuales.

## ▶ Servidores virtuales

- Apache soporta sitios o servidores virtuales basados en IP, nombres y puertos (se explican posteriormente).
  - Varios servidores sobre una misma instalación de Apache.
- `<VirtualHost> ... </VirtualHost>`

# Servidor “principal” vs Servidores virtuales

---

## ▶ *Linux (Debian/Ubuntu)*

- La versión precompilada (.deb) para *Debian/Ubuntu* tiene habilitados servidores virtuales.
- Tiene creado y habilitado un servidor virtual por defecto (*default*).

## ▶ *Windows*

- La versión de *Windows* no tiene habilitados por defecto los servidores virtuales.
- Se utiliza el servidor “principal”.

# Ficheros de configuración

## Introducción

---

- Configuración editando ficheros de texto.

```
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

- Web

- <http://httpd.apache.org/docs/2.4/configuring.html>

# Ficheros de configuración

## Linux (Debian/Ubuntu)

---

- Directorio:

- **/etc/apache2**

```
alumno@ServidorLinux01:/etc/apache2$ ls -l
total 80
-rw-r--r-- 1 root root 7115 ene 7 2014 apache2.conf
drwxr-xr-x 2 root root 4096 jun 8 12:33 conf-available
drwxr-xr-x 2 root root 4096 jun 8 12:33 conf-enabled
-rw-r--r-- 1 root root 1782 ene 3 2014 envvars
-rw-r--r-- 1 root root 31063 ene 3 2014 magic
drwxr-xr-x 2 root root 12288 jun 8 12:33 mods-available
drwxr-xr-x 2 root root 4096 jun 8 12:33 mods-enabled
-rw-r--r-- 1 root root 320 ene 7 2014 ports.conf
drwxr-xr-x 2 root root 4096 jun 8 12:33 sites-available
drwxr-xr-x 2 root root 4096 jun 8 12:33 sites-enabled
alumno@ServidorLinux01:/etc/apache2$ _
```

# Ficheros de configuración

## *Linux (Debian/Ubuntu)*

---

- ▶ **/etc/apache2/apache2.conf**
  - Fichero de configuración principal.
  - Contienen un conjunto de directivas que determinan el comportamiento del servidor.
  - Incluye (directiva include) otros ficheros de configuración.
- ▶ **/etc/apache2/ports.conf**
  - Incluido en apache2.conf.
  - Se definen las IPs y puertos donde escucha el servidor.

# Ficheros de configuración

## *Linux (Debian/Ubuntu)*

---

```
# Formato general apache2.conf

# Aquí empieza la Sección 1 (directivas de configuración global)

... Directivas globales

# Aquí empezaría la sección 2 (directivas de funcionamiento del
# servidor principal )
... Directivas de funcionamiento del servidor principal (se heredan en
los servidores virtuales)
User ${APACHE_RUN_USER}
Group ${APACHE_RUN_GROUP}

include /etc/apache2/mods-enabled/*.load
include /etc/apache2/mods-enabled/*.conf
include /etc/apache2/httpd.conf
include /etc/apache2/ports.conf

.. Directivas de logs y errores
include /etc/apache2/conf.d/

# Aquí empezaría la sección 3 (Servidores virtuales)
include /etc/apache2/sites-enabled/
```

# Ficheros de configuración

## *Linux (Debian/Ubuntu)*

---

### ► Directorios de configuraciones globales

- `/etc/apache2/conf-available/`

- Configuraciones disponibles.

```
alumno@ServidorLinux01:/etc/apache2/conf-available$ ls
charset.conf          other-vhosts-access-log.conf  serve-cgi-bin.conf
localized-error-pages.conf  security.conf
alumno@ServidorLinux01:/etc/apache2/conf-available$ _
```

- `/etc/apache2/conf-enabled/`

- Configuraciones habilitadas.
- Enlaces simbólicos a los ficheros de `conf-available`.

```
alumno@ServidorLinux01:/etc/apache2/conf-enabled$ ls
charset.conf          other-vhosts-access-log.conf  serve-cgi-bin.conf
localized-error-pages.conf  security.conf
alumno@ServidorLinux01:/etc/apache2/conf-enabled$ _
```

# Ficheros de configuración

## *Linux (Debian/Ubuntu)*

---

### ► Directorios de configuración de módulos

- `/etc/apache2/mods-available/`

- Módulos disponibles.

```
data.load          npm_worker.load    suexec.load
dav_fs.conf        negotiation.conf    unique_id.load
dav_fs.load        negotiation.load    userdir.conf
dav.load           proxy_ajp.load      userdir.load
dav_lock.load      proxy_balancer.conf usertrack.load
dbd.load           proxy_balancer.load vhost_alias.load
deflate.conf       proxy.conf          xml2enc.load
deflate.load       proxy_connect.load
alumno@ServidorLinux01:/etc/apache2/mods-available$
```

- `/etc/apache2/mods-enabled/`

- Módulos habilitados.
- Enlaces simbólicos a los ficheros de `mods-available`.

```
alumno@ServidorLinux01:/etc/apache2/mods-enabled$ ls
access_compat.load  authz_host.load    dir.load          negotiation.conf
alias.conf          authz_user.load    env.load          negotiation.load
alias.load          autoindex.conf     filter.load       setenvif.conf
auth_basic.load     autoindex.load     mime.conf         setenvif.load
authn_core.load     deflate.conf       mime.load         status.conf
authn_file.load     deflate.load        mpm_event.conf   status.load
authz_core.load     dir.conf           mpm_event.load
```

# Ficheros de configuración

## *Linux (Debian/Ubuntu)*

---

### ► Directorios de configuración de sitios (o servidores) virtuales

- **/etc/apache2/sites-available/**
  - Contienen el fichero `default` con la configuración del servidor virtual por defecto.

```
alumno@ServidorLinux01:/etc/apache2/sites-available$ ls  
000-default.conf  default-ssl.conf
```

- **/etc/apache2/sites-enabled/**
  - Contienen el fichero `000-default` que es un enlace al fichero `default` del directorio `sites-enable`.

```
alumno@ServidorLinux01:/etc/apache2/sites-enabled$ ls  
000-default.conf
```

# Ficheros de configuración

## *Linux (Debian/Ubuntu)*

---

- **/etc/apache2/envvars**
  - Define variables de entorno.

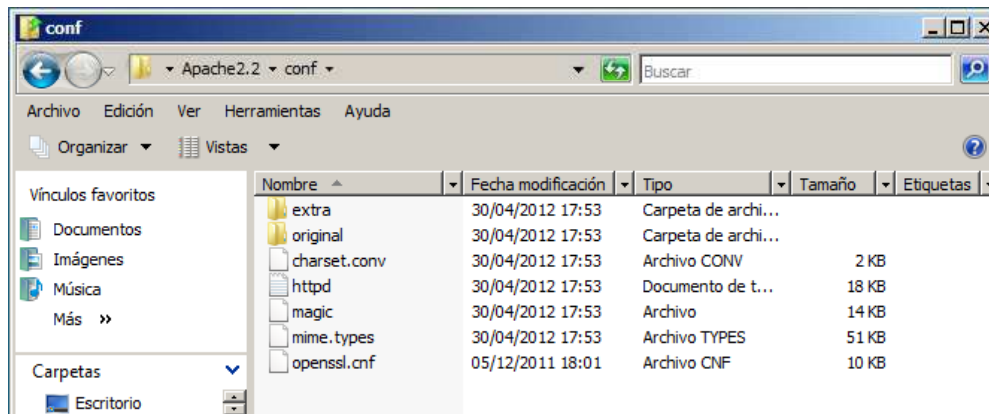
# Ficheros de configuración

## Windows

---

### ► Directorio:

- **C:\Program Files\Apache Software Foundation\Apache2.2\conf**



# Ficheros de configuración

## Windows

---

- **C:\Program Files\Apache Software Foundation\Apache2.2\conf\httpd.conf**
  - Fichero de configuración principal.
  - Contienen un conjunto de directivas que determinan el comportamiento del servidor.
  - Puede incluir (directiva `include`) otros ficheros de configuración.



# Directivas

- ▶ Se definen en los ficheros de configuración.
- ▶ Las directivas no especificadas utilizan su valor por defecto.
- ▶ Comentarios -> #
- ▶ Muchas directivas.
  - ▶ <http://httpd.apache.org/docs/2.4/mod/directives.html>
  - ▶ <http://httpd.apache.org/docs/2.4/mod/quickreference.html>

# Directivas

- [DefaultIcon](#)
- [DefaultLanguage](#)
- [DefaultType](#)
- [DeflateBufferSize](#)
- [DeflateCompressionLevel](#)
- [DeflateFilterNote](#)
- [DeflateMemLevel](#)
- [DeflateWindowSize](#)
- [Deny](#)
- [<Directory>](#)
- [DirectoryIndex](#)
- [<DirectoryMatch>](#)
- [DirectorySlash](#)
- [DocumentRoot](#)
- [DumpIOInput](#)
- [DumpIOLogLevel](#)
- [DumpIOOutput](#)
- [EnableExceptionHook](#)
- [EnableMMAP](#)

**<Directory> Directive**

**Description:** Enclose a group of directives that apply only to the named file-system directory, sub-directories, and their contents

**Syntax:** <Directory directory-path> ... </Directory>

**Context:** server config, virtual host

**Status:** Core

**Module:** core

<Directory> and </Directory> are used to enclose a group of directives that will apply only to the named directory, sub-directories of that directory, and the files within the respective directories. Any directive that is allowed in a directory context may be used. *Directory-path* is either the full path to a directory, or a wild-card string using Unix shell-style matching. In a wild-card string, ? matches any single character, and \* matches any sequences of characters. You may also use {} character ranges. None of the wildcards match a "/" character, so <Directory /\*/public\_html> will not match /home/user/public\_html, but <Directory /home/\*/public\_html> will match. Example:

```
<Directory /usr/local/httpd/htdocs>
Options Indexes FollowSymLinks
</Directory>
```

Be careful with the directory-path arguments: They have to literally match the filesystem path which Apache uses to access the files. Directives applied to a particular <Directory> will not apply to files accessed from that same directory via a different path, such as via different symbolic links.

Regular expressions can also be used, with the addition of the ~ character. For example:

```
<Directory ~ "*/www/.*/[0-9]{3}">
```

would match directories in /www/ that consisted of three numbers.

If multiple (non-regular expression) <Directory> sections match the directory (or one of its parents) containing a document, then the directives are applied in the order of

# Directivas

## ► Algunas de las que usaremos:

- ServerRoot
- Listen
- DocumentRoot
- DirectoryIndex
- <Directory ... > ... </Directory>
- <Files ... > ... </Files>
- Options Indexes All None
- Alias
- LoadModule
- <IfModule ... > ... </IfModule>
- ServerName

# Práctica

## ► Práctica 5.3

- Ficheros de configuración y directivas de *Apache 2.4* en *Linux*.

```
alumno@ServidorLinux01:/etc/apache2$ ls -l
total 80
-rw-r--r-- 1 root root 7115 ene 7 2014 apache2.conf
drwxr-xr-x 2 root root 4096 jun 8 12:33 conf-available
drwxr-xr-x 2 root root 4096 jun 8 12:33 conf-enabled
-rw-r--r-- 1 root root 1782 ene 3 2014 envvars
-rw-r--r-- 1 root root 31063 ene 3 2014 magic
drwxr-xr-x 2 root root 12288 jun 8 12:33 mods-available
drwxr-xr-x 2 root root 4096 jun 8 12:33 mods-enabled
-rw-r--r-- 1 root root 320 ene 7 2014 ports.conf
drwxr-xr-x 2 root root 4096 jun 8 12:33 sites-available
drwxr-xr-x 2 root root 4096 jun 8 12:33 sites-enabled
alumno@ServidorLinux01:/etc/apache2$
```

```
LogFormat "%h %l %u %t \"%r\"> %s %D \"%{Referer}i\" \"%{User-Agent}i\"> combi
LogFormat "%h %l %u %t \"%r\"> %s %D" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf

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

```
# VirtualHost *:80
#
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

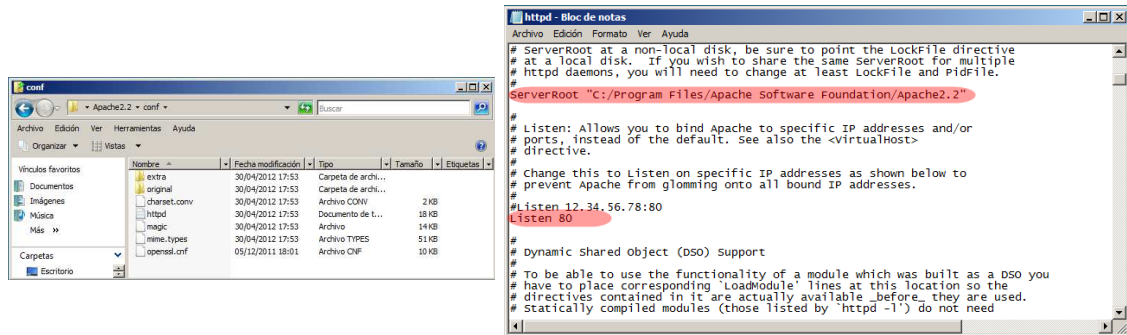
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# 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
```

# Práctica

## ► Práctica 5.4

- Ficheros de configuración y directivas de *Apache 2.2* en *Windows*.



# Configuración básica

## ► Ficheros a servir por defecto

- DirectoryIndex

```
# 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>
```

# Configuración básica

## ► Opciones sobre directorios

- `<Directory> ... </Directory>`
- Options Indexes

```
DocumentRoot /var/www
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>

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

<Directory /var/www/ciclos>
    Options FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>
```

Despliegue de aplicaciones web

39

# Configuración básica

## ► Códigos de error

- ErrorDocument

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

ErrorDocument 404 "Página no encontrada en la red daw01.net"

</VirtualHost>
```

```
ErrorLog ${APACHE_LOG_DIR}/error.log

# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn

CustomLog ${APACHE_LOG_DIR}/access.log combined

ErrorDocument 404 /404.html
```

Despliegue de aplicaciones web

40

# Configuración básica

---

## ► Directorios virtuales

- Directorios que están fuera del directorio raíz (DocumentRoot) del servidor.
- Alias

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

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

ErrorDocument 404 /404.html

</VirtualHost>
```

# Configuración básica

---

## ► Redirecciones

- Redirect

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

Redirect /fp http://www.todofp.es_

ErrorDocument 404 /404.html
```

# Práctica

## ► Práctica 5.5

- Configuración básica en *Linux*.

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

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

ErrorDocument 404 /404.html

</VirtualHost>
```

# Práctica

## ► Práctica 5.6 (1)

- Configuración básica en *Windows*

```
<Directory "C:/Program Files/Apache Software Foundation/Apache2.2/htdocs">
#
# Possible values for the Options directive are "None", "All",
# or any combination of:
#   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.
#
# The Options directive is both complicated and important. Please see
# http://httpd.apache.org/docs/2.2/mod/core.html#options
# for more information.
#
DirectoryIndex despliegue.html
Options Indexes FollowSymLinks

#
# AllowOverride controls what directives may be placed in .htaccess files.
# It can be "All", "None", or any combination of the keywords:
#   Options FileInfo AuthConfig Limit
#
AllowOverride None

#
# Controls who can get stuff from this server.
#
Order allow,deny
Allow from all

</Directory>
```

# Práctica

## ► Práctica 5.6 (2)

### ◦ Configuración básica en *Windows*

```
<Directory "C:/Program Files/Apache Software Foundation/Apache2.2/htdocs/ciclos">
    Options FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

Alias /apuntes C:/Users/Administrador/apuntes
<Directory "C:/Users/Administrador/apuntes">
    Options Indexes FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

# Módulos

## Introducción

## ► Ampliación del núcleo de Apache con módulos.

## ► Webs

- <https://modules.apache.org/>
- <http://httpd.apache.org/docs/2.4/mod/>

Funcionalidad Básica y Módulos de MultiProcesamiento

core

Core Apache HTTP Server features that are always available

mpm\_common

A collection of directives that are implemented by more than one multi-processing module (MPM)

beos

Este módulo de multiprocesamiento está optimizado para BeOS

event

An experimental variant of the standard [worker](#) MPM

mpm\_network

Multi-Processing Module implementing an exclusively threaded web server optimized for Novell NetWare

mpm\_os2

Hybrid multi-process, multi-threaded MPM for OS/2

prefork

Implements a non-threaded, pre-forking web server

mpm\_winnt

This Multi-Processing Module is optimized for Windows NT.

worker

Multi-Processing Module implementing a hybrid multi-threaded multi-process web server

Otros Módulos

A C D E F H I L M N P R S U V

Last 20 modules modified/added

Apache Mobile Filter  
mod\_vlmiti  
mod\_process\_security  
mod\_lalimit  
mod\_rchecker  
mod\_vlimitconn  
mod\_xml2  
mod\_ruid2  
DACS  
mod\_badge  
mod\_proxy\_filter\_xff  
Apache Rivet  
mod\_limits  
mod\_auth\_useragent2  
mod\_rangelimit  
mod\_chxj  
mod\_fortune  
mod\_removepass  
mod\_access\_dnsbl  
mod\_baik

Total Module Count: 540

# Módulos

## Introducción

- ▶ Cada módulo
  - Funcionalidades.
  - Directivas para configurarlas.

### Apache Module mod\_alias

Available Languages: [en](#) | [ja](#) | [ko](#) | [tr](#)

<b>Description:</b>	Provides for mapping different parts of the host filesystem in the document tree and for URL redirection
<b>Status:</b>	Base
<b>Module Identifier:</b>	alias_module
<b>Source File:</b>	mod_alias.c

**Summary**

The directives contained in this module allow for manipulation and control of URLs as requests arrive at the server. The [Alias](#) and [ScriptAlias](#) directives are used to map between URLs and filesystem paths. This allows for content which is not directly under the [DocumentRoot](#) served as part of the web document tree. The [ScriptAlias](#) directive has the additional effect of marking the target directory as containing only CGI scripts.

The [Redirect](#) directives are used to instruct clients to make a new request with a different URL. They are often used when a resource has moved to a new location.

[mod\\_alias](#) is designed to handle simple URL manipulation tasks. For more complicated tasks such as manipulating the query string, use the tools provided by [mod\\_rewrite](#).

**Directives**

- [Alias](#)
- [AliasMatch](#)
- [Redirect](#)
- [RedirectMatch](#)
- [RedirectPermanent](#)
- [RedirectTemp](#)
- [ScriptAlias](#)
- [ScriptAliasMatch](#)

**Topics**

- [Order of Processing](#)

**See also**

- [mod\\_rewrite](#)
- [Mapping URLs to the filesystem](#)

# Módulos

## Introducción

- ▶ Tipos
  - **Módulos estáticos** que se añaden cuando se compila *Apache*.
  - **Módulos** que se cargan **dinámicamente** cuando se inicia el servidor.
    - Hay que compilar el servidor con la opción DSO (*Dynamic Shared Object*).
      - Ventajas
        - Servidor más flexible.
        - Más sencillo el prototipado y desarrollo de módulos.
      - Desventajas DSO
        - Servidor es más lento en el arranque.
        - Servidor más lento en funcionamiento.



# Módulos

## Introducción

---

### ► Directivas

- **LoadModule**
  - Permite cargar módulos dinámicos.
- **<IfModule nombre\_modulo> ... </IfModule>**
  - Especificar directivas que se tendrán en cuenta si el módulo está cargado.

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

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

# Módulos

## *Linux (Debian/Ubuntu)*

---

### ► Módulos disponibles

- **/usr/lib/apache2/modules**

```
alumno@ServidorLinux01:/usr/lib/apache2/modules$ ls
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_scgi.so
mod_authn_dbm.so mod_dumpio.so   mod_proxy.so
mod_authn_default.so mod_env.so      mod_reqtimeout.so
mod_authn_file.so mod_expires.so  mod_rewrite.so
mod_authnz_ldap.so mod_ext_filter.so mod_setenvif.so
mod_authz_dbm.so mod_file_cache.so mod_speling.so
mod_authz_default.so mod_filter.so   mod_ssl.so
mod_authz_groupfile.so mod_headers.so  mod_status.so
mod_authz_host.so mod_ident.so    mod_substitute.so
mod_authz_owner.so mod_imagemap.so mod_suexec.so
mod_authz_user.so mod_include.so  mod_unique_id.so
mod_autoindex.so mod_info.so     mod_userdir.so
mod_cache.so   mod_ldap.so     mod_usertrack.so
mod_cern_meta.so mod_log_forensic.so mod_version.so
mod_cgid.so    mod_mem_cache.so mod_vhost_alias.so
```

# Módulos

## *Linux (Debian/Ubuntu)*

---

### ► Directorios y ficheros de configuración (1)

- `/etc/apache2/mods-available/`
  - Módulos disponibles.
  - Ficheros `.load`
    - Para cargar un módulo.
  - Ficheros `.conf`
    - Configuración básica para iniciar el módulo.

```
alumno@ServidorLinux01:/etc/apache2/mods-available$ ls
actions.conf      cern_meta.load    ident.load         proxy_http.load
actions.load      cgid.conf          imagemap.load      proxy.load
alias.conf         cgid.load          include.load        proxy_scgi.load
alias.load         cgi.load           info.conf           reqtimeout.conf
```

# Módulos

## *Linux (Debian/Ubuntu)*

---

### ► Directorios y ficheros de configuración (2)

- `/etc/apache2/mods-enabled/`
  - Módulos habilitados.
  - Enlaces simbólicos a los ficheros de `mods-available`.
  - Módulos a cargar al iniciar Apache.

```
lrwxrwxrwx 1 root root 26 abr 30 14:23 dir.conf -> ../mods-available/dir.conf
lrwxrwxrwx 1 root root 26 abr 30 14:23 dir.load -> ../mods-available/dir.load
lrwxrwxrwx 1 root root 26 abr 30 14:23 env.load -> ../mods-available/env.load
lrwxrwxrwx 1 root root 27 abr 30 14:23 mime.conf -> ../mods-available/mime.conf
lrwxrwxrwx 1 root root 27 abr 30 14:23 mime.load -> ../mods-available/mime.load
```

# Módulos

## Linux (Debian/Ubuntu)

### ► Comandos

- Habilitar un módulo
  - `a2enmod nombre_modulo`
    - Crea un enlace simbólico en `mods_enabled`.
- Deshabilitar un módulo
  - `a2dismod nombre_modulo`
    - Borra el enlace simbólico de `mods_enabled`.

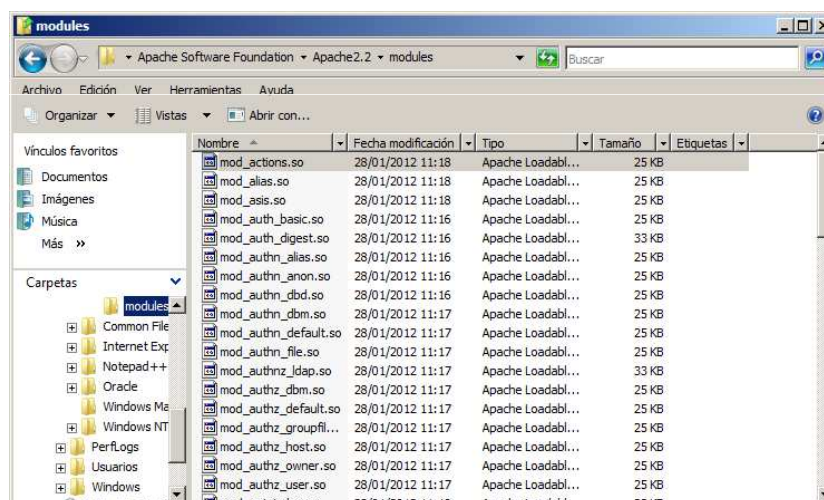
- Hay que reiniciar Apache2 al habilitar /deshabilitar módulos.

# Módulos

## Windows

### ► Módulos disponibles

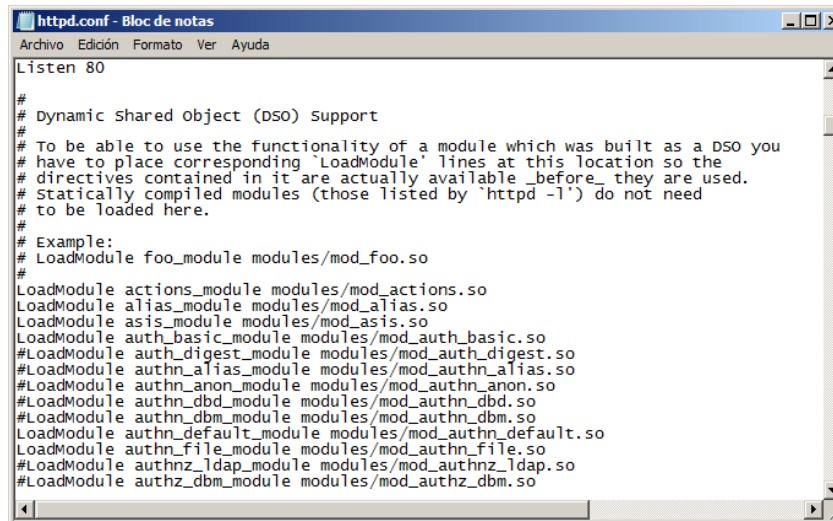
- C:\Program Files\Apache Software Foundation\Apache2.2\modules



# Módulos

## Windows

- ▶ **Habilitar/deshabilitar módulos.**
  - **httpd.conf**



```
httpd.conf - Bloc de notas
Archivo  Edición  Formato  Ver  Ayuda
Listen 80

#
# Dynamic Shared Object (DSO) support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by 'httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule actions_module modules/mod_actions.so
LoadModule alias_module modules/mod_alias.so
LoadModule asis_module modules/mod_asis.so
LoadModule auth_basic_module modules/mod_auth_basic.so
LoadModule auth_digest_module modules/mod_auth_digest.so
LoadModule authn_alias_module modules/mod_authn_alias.so
LoadModule authn_anon_module modules/mod_authn_anon.so
LoadModule authn_dbd_module modules/mod_authn_dbd.so
LoadModule authn_dbm_module modules/mod_authn_dbm.so
LoadModule authn_default_module modules/mod_authn_default.so
LoadModule authn_file_module modules/mod_authn_file.so
LoadModule authnz_ldap_module modules/mod_authnz_ldap.so
LoadModule authz_dbm_module modules/mod_authz_dbm.so
```

## Práctica

- ▶ **Práctica 5.7**
  - **Módulos en *Linux*.**

```
LoadModule alias_module /usr/lib/apache2/modules/mod_alias.so
```

```
<IfModule alias_module>
# Aliases: Add here as many aliases as you need (with no limit). The full
# syntax is: Alias fakedname realname
#
# Note that if you include a trailing / on fakedname then the server will
# require it to be present in the URL. So "/icons" isn't aliased in this
# example, only "/icons/". If the fakedname is slash-terminated, then the
# realname must also be slash terminated, and if the fakedname omits the
# trailing slash, the realname must also omit it.
#
# We include the /icons/ alias for FancyIndexed directory listings. If
# you do not use FancyIndexing, you may comment this out.

Alias /icons/ "/usr/share/apache2/icons/"

<Directory "/usr/share/apache2/icons">
    Options FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
</IfModule>
```

# Práctica

## ► Práctica 5.8

### ◦ Módulos en *Windows*.

```
LoadModule userdir_module modules/mod_userdir.so

# Settings for user home directories
# Required module: mod_userdir

#
# UserDir: The name of the directory that is appended onto a user's home
# directory if a ~user request is received. Note that you must also set
# the default access control for these directories, as in the example below.
#
UserDir "My Documents/My Website"

#
# Control access to UserDir directories. The following is an example
# for a site where these directories are restricted to read-only.
#
<Directory "C:/Users/*/My Documents/My Website">
    AllowOverride FileInfo AuthConfig Limit Indexes
    Options Multiviews Indexes SymLinksIfOwnerMatch IncludesNoExec
    <Limit GET POST OPTIONS>
        Order allow,deny
        Allow from all
    </Limit>
    <LimitExcept GET POST OPTIONS>
        Order deny,allow
        Deny from all
    </LimitExcept>
</Directory>
```

# Control de acceso

## ► Control de acceso a recursos: ficheros, directorios, URLs, ...

### ◦ Control de acceso por host (IP/nombre\_dominio)

- Módulo [mod\\_authz\\_host](#).

### ◦ Control de acceso por variables de entorno

- Módulo [mod\\_authz\\_host](#).
- Módulo [mod\\_setenvif](#).

### ◦ Control de acceso usando el módulo rewrite

- Módulo [mod\\_rewrite](#).

## ► Web

- <http://httpd.apache.org/docs/2.4/es/howto/access.html>

# Control de acceso

---

## ► Apache 2.2 (1)

### ◦ Control de acceso (IP/nombre\_dominio) (1)

```
<Directory /var/www/profesor>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from 127.0.0.1
    allow from 192.168.1.16
</Directory>
```

- Order Deny,Allow | Allow,Deny
- Allow from
- Deny from

# Control de acceso

---

## ► Apache 2.2 (2)

### ◦ Control de acceso (IP/nombre\_dominio) (2)

- Order Deny,Allow
  - **El acceso está permitido por defecto.** Las directivas Deny se evalúan antes que las directivas Allow. Cualquier cliente que “no case” con una directiva Deny a Allow tendrá permitido el acceso. Si el cliente “casa” al mismo tiempo en una directiva Allow y otra Deny, tendrá permitido el acceso por que las directivas Allow se evalúan las últimas.
- Order Allow,Deny
  - **El acceso está denegado por defecto.** Las directivas Allow se evalúan antes que las directivas Deny. Cualquier cliente que “no case” con una directiva Deny a Allow tendrá denegado el acceso. Si el cliente “casa” al mismo tiempo en una directiva Allow y otra Deny, tendrá denegado el acceso por que las directivas Deny se evalúan las últimas.

# Control de acceso

## ► Apache 2.2 (3)

### ◦ Control de acceso (IP/nombre\_dominio) (3)

#### • Ejemplos (1)

Order Deny,Allow	1) Acceso permitido por defecto
Deny from all	2) Todos los hosts so denegados
Allow from daw.org	3) Se permite el acceso a los hosts de dominio *.daw.org

Resultado: Solo los host de \*.daw.org son permitidos.

Order Allow, Deny	1) Acceso denegado por defecto
Allow from daw.org	2) Se permite el acceso a los hosts de dominio *.daw.org
Deny from bbdd.daw.org	3) Se deniega el acceso a los hosts de dominio *.bbdd.daw.org

Resultado: Los hosts de \*.daw.org son permitidos execepto los de \*.bbdd.daw.org.  
¿Qué ocurre si se cambia el orden a Order Deny,Allow ?

# Control de acceso

## ► Apache 2.2 (4)

### ◦ Control de acceso (IP/nombre\_dominio) (4)

#### • Ejemplos (2)

Order Allow,Deny
Allow from 200.200.100.0/24
Deny from www.daw.org

¿Resultado?

Order Allow,Deny
Allow from 192.168.0.0/16
Deny from all

¿Resultado?

# Control de acceso

---

## ► Apache 2.4 (1)

### ◦ Control de acceso (IP/nombre\_dominio) (1)

```
<Directory /var/www/html/profesor>  
    Options Indexes FollowSymLinks  
    AllowOverride None  
    Require ip 127.0.0.1  
    Require ip 192.168.1.16  
</Directory>
```

- Require
- <http://httpd.apache.org/docs/2.4/upgrading.html>

# Control de acceso

---

## ► Apache 2.4 (2)

### ◦ Control de acceso (IP/nombre\_dominio) (2)

- Por defecto se deniega todo.
- Se permite con la directiva **Require**.
- [http://httpd.apache.org/docs/2.4/mod/mod\\_authz\\_host.html](http://httpd.apache.org/docs/2.4/mod/mod_authz_host.html)



# Control de acceso

---

## ► Apache 2.4 (3)

### ◦ Control de acceso (IP/nombre\_dominio) (3)

- Ejemplos (1) In this example, all requests are denied.

#### 2.2 configuration:

```
Order deny,allow
Deny from all
```

#### 2.4 configuration:

```
Require all denied
```

In this example, all requests are allowed.

#### 2.2 configuration:

```
Order allow,deny
Allow from all
```

#### 2.4 configuration:

```
Require all granted
```

# Control de acceso

---

## ► Apache 2.4 (4)

### ◦ Control de acceso (IP/nombre\_dominio) (4)

- Ejemplos (2)

In the following example, all hosts in the example.org domain are allowed access; all other hosts are denied access.

#### 2.2 configuration:

```
Order Deny,Allow
Deny from all
Allow from example.org
```

#### 2.4 configuration:

```
Require host example.org
```

# Práctica

---

## ► Práctica 5.9

- Control de acceso por IP y nombre de dominio.

```
<Directory /var/www/html/profesor>  
    Options Indexes FollowSymLinks  
    AllowOverride None  
    Require ip 127.0.0.1  
    Require ip 192.168.1.16  
</Directory>
```

# Autenticación y autorización

---

## ► Autenticación

- Proceso para verificar que alguien es realmente quien dice ser.

## ► Autorización

- Proceso por el que se permite a alguien hacer o acceder a algo que quiere.

# Autenticación y autorización

---

## ► Tipos de autenticación

- *Basic*
  - Módulo `mod_auth_basic`.
- *Digest*
  - Módulo `mod_auth_digest`.

## ► *Formularios HTTL.*

## ► *Certificados digitales*

- Módulo `mod_ssl`.

## ► Web

- <http://httpd.apache.org/docs/2.4/es/howto/auth.html>

# Autenticación y autorización

---

## ► Proveedores de autenticación

- Módulos que ofrecen la posibilidad de acceder a credenciales (usuarios, contreras, certificados, ...) usados en la autenticación en:
  - Ficheros de texto.
  - Bases de datos.
  - Servidores de directorios (LDAP).
  - ...

## ► Web

- <http://httpd.apache.org/docs/2.4/es/howto/auth.html>

# Autenticación y autorización

---

## ► Módulos de autorización

- Módulos que permite realizar el proceso de autorización sobre:
  - Ficheros de texto.
  - Bases de datos.
  - Servidores de directorios (LDAP).
  - ...

## ► Web

- <http://httpd.apache.org/docs/2.4/es/howto/auth.html>
- <http://httpd.apache.org/docs/2.4/es/mod/core.html#require>

# Autenticación y autorización

---

## ► Autenticación *Basic* (1)

- [mod\\_auth\\_basic](#)
- La contraseña es enviada por el cliente en texto plano.
- Autenticación y autorización sobre fichero de texto (htpasswd).
  - [mod\\_authn\\_file](#)
  - [mod\\_authz\\_user](#)

# Autenticación y autorización

## ► Autenticación *Basic* (2)

- 1) Crear fichero con usuarios/contraseñas
  - Htpasswd
    - Hay que instalar el paquete **apache2-utils**
    - <http://httpd.apache.org/docs/2.4/es/programs/htpasswd.html>

```
# La primera vez que se invoca el comando se
# utiliza a opción -c para crear el fichero
htpasswd -c /etc/apache2/passwd profesor1

# Añade un nuevo usuario al fichero
htpasswd /etc/apache2/passwd profesor2

# Borrar un nuevo usuario al fichero
htpasswd -D /etc/apache2/passwd profesor1
```

# Autenticación y autorización

## ► Autenticación *Basic* (3)

- 2) Definir directivas

```
<Directory /var/www/html/profesor>
  Options Indexes FollowSymLinks
  AllowOverride None
  AuthType Basic
  AuthName "Acceso restringido"
  AuthUserFile /etc/apache2/passwd
  <RequireAll>
    Require user profesor1 profesor2
    <RequireAny>
      Require ip 127.0.0.1
      Require ip 192.168.1.16
    </RequireAny>
  </RequireAll>
</Directory>
```

- AuthType, AuthName, AuthUserFile, AuthGroupFile, Require, ...

# Autenticación y autorización

---

- ▶ Las directivas <RequireAll>, <RequireAny> and <RequireNone> se pueden combinar con la directiva Require para controlar que lógica de autorización se utilizará
  - [http://httpd.apache.org/docs/2.4/mod/mod\\_authz\\_core.html](http://httpd.apache.org/docs/2.4/mod/mod_authz_core.html)

# Autenticación y autorización

---

- ▶ Autenticación *Digest* (1)
  - mod\_auth\_digest
  - La contraseña se envía cifrada (¡¡ **cifrado débil, no es seguro !!**) por el cliente.
  - Autenticación y autorización sobre fichero de texto (htdigest).
    - mod\_auth\_digest
    - mod\_authz\_user

# Autenticación y autorización

---

## ► Autenticación *Digest* (2)

- 1) Crear fichero con usuarios/contraseñas asociados a un dominio (*realm*).
  - **htdigest**
    - <http://httpd.apache.org/docs/2.4/es/programs/htdigest.html>

```
# La primera vez que se invoca el comando se
# utiliza a opción -c para crear el fichero
htdigest -c /etc/apache2/passwd informatica admin1

# Añade un nuevo usuario al fichero
Htdigest /etc/apache2/passwd informatica admin2

# Borrar un nuevo usuario al fichero
htdigest -D /etc/apache2/passwd informatica admin1
```

# Autenticación y autorización

---

## ► Autenticación *Digest* (3)

- 2) Definir directivas

```
<Directory /var/www/html/departamento>
    Options Indexes FollowSymLinks
    AllowOverride None
    AuthType Digest
    AuthName "informatica"
    AuthDigestProvider file
    AuthUserFile /etc/apache2/digest
    Require user admin1 admin2
</Directory>
```

- AuthType, AuthName, AuthDigestProvider, AuthUserFile, AuthGroupFile, Require, ...

# Práctica

## ► Práctica 5.10

- Autenticación y autorización *Basic* y *Digest*

```
<Directory /var/www/profesor>
  Options Indexes FollowSymLinks MultiViews
  AllowOverride None
  Order allow,deny
  allow from 127.0.0.1
  allow from 192.168.1.16
  AuthType Basic
  AuthName "Acceso restringido"
  AuthUserFile /etc/apache2/passwd
  Require user profesor1 profesor2
</Directory>
```

```
<Directory /var/www/departamento>
  Options Indexes FollowSymLinks MultiViews
  AllowOverride None
  AuthType Digest
  AuthName "informatica"
  AuthDigestProvider file
  AuthUserFile /etc/apache2/digest
  Require user admin1 admin2
</Directory>
```

# Ficheros .htaccess

- Ficheros que permiten la configuración personalizada de directorios.

- Fichero de configuración de *Apache*.

```
Alias /blog /home/profesor/blog
<Directory /home/profesor/blog>
  AllowOverride All
</Directory>
```

- Fichero .htaccess dentro de un directorio.

```
Options Indexes
AuthType Digest
AuthName "informatica"
AuthDigestProvider file
AuthUserFile /home/profesor/blog/.htdigest
Require user blog
```



# Ficheros .htaccess

---

- ▶ Cada vez que se produce una petición:
  - El servidor busca en la ruta del recurso que ha solicitado el cliente un fichero con el nombre .htaccess.
  - Aplica sobre el directorio las directivas definidas.
- ▶ En la configuración del servidor hay que permitir el uso de estos ficheros.
- ▶ Web
  - <http://httpd.apache.org/docs/2.4/es/howto/htaccess.html>

# Ficheros .htaccess

---

- ▶ Definida la siguiente directiva a nivel del servidor principal para que los ficheros que empiecen con .ht no sea visibles por los clientes.

- Windows

```
<FilesMatch "^\.ht">  
    Order allow,deny  
    Deny from all  
    Satisfy All  
</FilesMatch>
```

- Linux

```
<Files ~ "^\.ht">  
    Order allow,deny  
    Deny from all  
    Satisfy all  
</Files>
```

# Ficheros .htaccess

---

- ▶ No se deben usar a menos que no se tenga acceso al archivo de configuración del servidor (Ej.: Servidor de *hosting*)
  - Eficiencia.
  - Seguridad.
- ▶ El nombre .htaccess se puede cambiar con la directiva `AccessFileName`.

## Práctica

---

- ▶ **Práctica 5.1.1**
  - Ficheros .htaccess.

```
Alias /blog /home/profesor/blog
<Directory /home/profesor/blog>
    AllowOverride All
</Directory>
```

```
Options Indexes
AuthType Digest
AuthName "informatica"
AuthDigestProvider file
AuthUserFile /home/profesor/blog/.htdigest
Require user blog
```

# Monitorización

## Ficheros de registros (*logs*)

---

- ▶ Linux (*Debian/Ubuntu*)
  - Errores
    - `/var/log/apache2/error.log`
  - Accesos
    - `/var/log/apache2/access.log`
- ▶ Windows
  - Errores
    - `C:\Program Files\Apache Software Foundation\Apache2.2\log/error.log`
  - Accesos
    - `C:\Program Files\Apache Software Foundation\Apache2.2\log/access.log`

# Monitorización

## Ficheros de registros (*logs*)

---

- ▶ Múltiples directivas para personalizar *logs*.
  - ErrorLog
  - LogLevel
  - CustomLog
  - LogFormat
  - ...
- ▶ **!!! Cuidado !!!**
  - Apache tiene que tener permisos para escribir en el directorio donde se guardan los ficheros de *logs*.
- ▶ Web
  - <http://httpd.apache.org/docs/2.4/en/logs.html>

# Práctica

---

## ► Práctica 5.12

- Ficheros de registros (*logs*).

```
ErrorLog ${APACHE_LOG_DIR}/error.log

# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn
```

```
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

# Monitorización

## Módulos para monitorizar el servidor

---

### ► mod\_status

- [http://httpd.apache.org/docs/2.4/mod/mod\\_status.html](http://httpd.apache.org/docs/2.4/mod/mod_status.html)

### ► mod\_info

- [http://httpd.apache.org/docs/2.4/mod/mod\\_info.html](http://httpd.apache.org/docs/2.4/mod/mod_info.html)

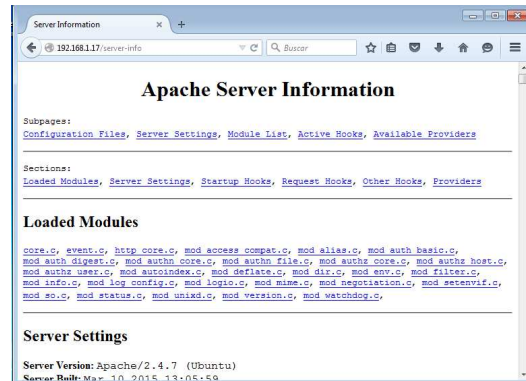
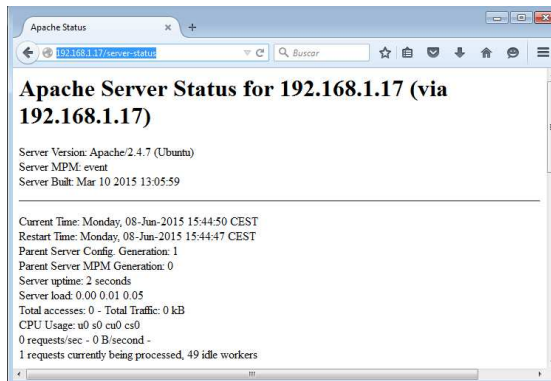
# Práctica

## ► Práctica 5.13

- Módulos *mod\_status* y *mod\_info*.

```
<Location /server-status>
    SetHandler server-status
    Require local
    Require ip 192.168.1.16
    #Require ip 192.0.2.0/24
</Location>
```

```
<Location /server-info>
    SetHandler server-info
    Require local
    Require ip 192.168.1.16
    #Require ip 192.0.2.0/24
</Location>
```



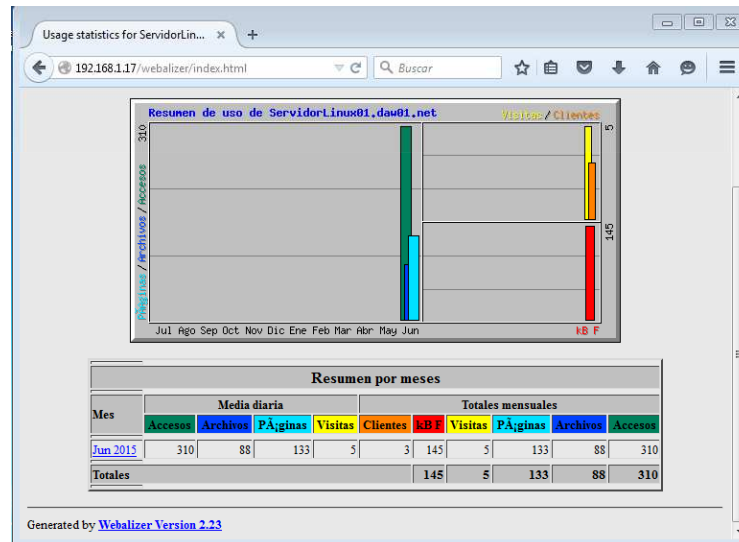
## Monitorización Analizadores de *logs*

- *Webalizer* (<http://www.webalizer.org/>)
- *Awstats* (<http://www.awstats.org/>)
- *Visitors* (<http://www.hping.org/visitors/>)
- *Analog* (<http://www.analog.cx/>)
- ...

# Práctica

## ► Práctica 5.14

### ◦ *Webalizer*



Despliegue de aplicaciones web

91

# Bibliografía

- Servicios de Red e Internet. Álvaro García Sánchez, Luis Enamorado Sarmiento, Javier Sanz Rodríguez. Editorial Garceta.
- <http://www.w3c.org>
- <http://httpd.apache.org>
- <http://www.apache.org>

Despliegue de aplicaciones web

92