

# Servidor Web Apache

Álvaro González Sotillo

Rey Fernando VI

12 de noviembre de 2015



- 1 Introducción
- 2 Configuración
  - Sitios
  - Directivas
  - Módulos
- 3 Control de acceso
  - Autorización
  - Autenticación
- 4 .htaccess
- 5 Monitorización



# Introducción

- <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.
- El servidor web de Apache es solo uno de los proyectos



**The Apache Software Foundation**

*Community-led development since 1999.*

# Servidor web Apache

- `http://httpd.apache.org`
- Servidor Web potente, flexible y ajustado al HTTP/1.1
- Modular.
- Altamente configurable y extensible
- Servidor “Open Source”
- Multiplataforma: Windows, Linux, Unix...
- Actualmente, el **más usado en Internet**

# Instalación

- Versiones:
  - 2.4, 2.2, 2.0, 1.3...
- Tipos de Instalación
  - Compilado desde los fuentes
  - **Utilizar paquetes binarios:** Por ejemplo, con `apt-get install Apache2`

## Instalación de *Apache2* en *Debian*

Instala *Apache2* en *Debian*. Comprueba que está instalado desde un navegador en la máquina real, visitando la dirección `http://192.168.3.200+n`

# Control del servidor

- En Linux
  - Parada del servidor:  
`service apache2 stop`
  - Arranque del servidor:  
`service apache2 start`
  - Volver a leer la configuración:  
`service apache2 reload`
- En Windows: Se instala como un servicio
- En cualquier sistema operativo
  - Utilizando el programa `apache2ctl`

# Control del servidor

- También puede arrancarse manualmente con el comando `apache2`
- En ese caso, no es un servicio/demonio
- Al cerrar el terminal/CTRL-C termina su ejecución

# Configuración

- Se realiza mediante ficheros de texto, en el directorio /etc/apache2
- <http://httpd.apache.org/docs/2.4/configuring.html>

```
root@debian8:/etc/apache2# ls -la
total 88
drwxr-xr-x  8 root root 4096 Oct  8 19:48 .
drwxr-xr-x 117 root root 4096 Oct  8 19:48 ..
-rw-r--r--  1 root root 7115 Aug  7 23:33 apache2.conf
drwxr-xr-x  2 root root 4096 Oct  8 19:48 conf-available
drwxr-xr-x  2 root root 4096 Oct  8 19:48 conf-enabled
-rw-r--r--  1 root root 1782 Aug  7 23:33 envvars
-rw-r--r--  1 root root 31063 Aug  2 00:21 magic
drwxr-xr-x  2 root root 12288 Oct  8 19:48 mods-available
drwxr-xr-x  2 root root 4096 Oct  8 19:48 mods-enabled
-rw-r--r--  1 root root  320 Aug  7 23:33 ports.conf
drwxr-xr-x  2 root root 4096 Oct  8 19:48 sites-available
drwxr-xr-x  2 root root 4096 Oct  8 19:48 sites-enabled
```

**Listado 1:** Contenido del directorio /etc/apache2



# Configuración

- `apache2.conf`
  - Fichero de configuración principal.
  - Opciones globales: logs, usuario de sistema, permisos en algunos directorios..
  - Incluye (directiva `Include` e `IncludeOptional`) otros ficheros de configuración.
- `ports.conf`
  - Direcciones IP y puertos donde escucha el servidor

# Configuración

- **envvars**
  - Define variables de entorno.
  - Es un script de shell lanzado antes que **apache**
- **httpd.conf**
  - En la mayoría de los sistemas se utiliza un fichero denominado httpd.conf como fichero de configuración principal.
  - Existe por compatibilidad y está vacío.

# Configuración: Sitios

- Los sitios configuran qué directorios son servidos
  - URL base de cada directorio
  - Permisos asociados
  - Dirección IP, nombre de servidor y puerto TCP
- Se encuentran en:
  - Directorio `sites-available`: Todos los sitios disponibles
  - Directorio `sites-enabled`: Sitios habilitados
  - Los sitios habilitados son enlaces a los sitios disponibles (como los módulos)
- Habilitar sitios: `a2ensite`
- Desabilitar sitios: `a2dissite`
- Tras modificar los sitios, es necesario reiniciar/recargar el servidor

# Directivas

- Los ficheros de configuración contienen directivas:
  - `http://httpd.apache.org/docs/2.4/mod/directives.html`
  - `http://httpd.apache.org/docs/2.4/mod/quickreference.html`
- Comentarios: `#`
- Una directiva no especificada tiene su valor por defecto

# Directivas

- Algunas directivas que usaremos
  - Algunas de las que usaremos:
  - ServerRoot
  - Listen
  - DocumentRoot
  - DirectoryIndex
  - Directory
  - Files
  - Options Indexes All None
  - Alias
  - LoadModule
  - IfModule
  - ServerName





# Ejercicios

## Directorio raíz de **apache**

Consulta el fichero de sitio `000-default` y deduce en qué directorio se aloja la página por defecto del sitio.

Modifica esa página para que su título sea “Servidor de *tu nombre*”

# Directivas: Ficheros por defecto

- Se definen dentro de `<Directory>` 
  - `DirectoryIndex`: Fichero a servir si se pide ese directorio
  - `Options Index` : Opción para habilitar listado de ficheros

## Directorio con listado

Crea el directorio `/var/www/html/despliegue`, y crea algunos ficheros dentro. Configura **apache** para que liste los ficheros al acceder a `http://192.168.200+n/despliegue`

## Directorio con página inicial

Crea el directorio `/var/www/html/daw`, y dentro crea un fichero `inicio.html`. Configura **apache** para que sirva el fichero `inicio.html` al acceder a `http://192.168.200+n/daw`

# Directivas: Errores

- **ErrorDocument**: Define qué página devuelve el servidor ante una condición de error

`ErrorDocument "número de error" "mensaje o página"`



## Directorio prohibido

Crea el directorio `/var/www/html/restringido`. Configura **apache** para que no liste los ficheros al acceder a él, y que muestre *Prohibido* en los errores 403. Accede a `http://192.168.200+n/restringido` y comprueba la configuración.

## No encontrado



Crea la página `/var/www/html/noencontrado.html`. Sírvela cuando se produzca un error 404.



# Directivas: Alias

- Se utiliza para “montar” directorios dentro de las URL servidas por Apache

Alias `"/camino-via-web"` *"directorio"*

```
Alias /apuntes /home/alumno/apuntes-despliegue
<Directory /home/alumno/apuntes>
  Options Indexes
  # FICHERO .htaccess (EXPLICADO DESPUES)
  AllowOverride None
  # PERMISOS (EXPLICADOS DESPUES)
  Require All Granted
</Directory>
```

**Listado 2:** Servir el directorio de apuntes en `http://192.168.3.200+n/apuntes`

# Directivas: Redirect

- Se utiliza para redirigir una petición a otra URL (respuesta 302 por defecto)

Redirect *"/camino-via-web"* *"URL"*

```
Redirect /fp http://www.todofp.es
```

**Listado 3:** Crear una redirección

# Introducción a los módulos

- Ampliaciones (plugins) de *Apache*
- Cada módulo aporta
  - Nueva funcionalidad
  - Nuevas directivas para ser configurada
- Tipos de Módulos
  - Estáticos: Incluidos al compilar *Apache*
  - Dinámicos: Librerías dinámicas enlazadas al iniciar *Apache*
    - Más lentos inicialmente
    - Más flexibilidad

# Directivas

- LoadModule: Carga un Módulos

`LoadModule nombremodulo Librería-DLL-o-SO`

- IfModule: Ejecución condicional si está cargado un módulo

`<IfModule nombremodulo> .... </IfModule>`

# Configuración: Módulos

- Los módulos son extensiones de funcionalidad de **apache** (plugins)
  - Directorio `mods-available`: Todos los módulos disponibles
  - Directorio `mods-enabled`: Módulos habilitados
  - Los módulos habilitados son enlaces a los módulos disponibles. Es posible habilitar y deshabilitar módulos modificando los enlaces con `ln -s`
- Habilitar módulos: `a2enmod`
- Desabilitar módulos: `a2dismod`
- Tras modificar los módulos, es necesario reiniciar/recargar el servidor

# Ejercicio con módulos

## Alias

Crea un alias en tu configuración de *Apache*. Después, activa y desactiva el módulo `alias`, y comprueba su resultado.

```
Alias /apuntes /home/alumno/apuntes-despliegue
<Directory /home/alumno/apuntes>
  Options Indexes
  # FICHERO .htaccess (EXPLICADO DESPUES)
  AllowOverride None
  # PERMISOS (EXPLICADOS DESPUES)
  Require All Granted
</Directory>
```

**Listado 4:** Creación de alias

# Control de acceso

- Filtro de acceso a URL, teniendo en cuenta
  - Recurso accedido
  - Host origen de la petición (IP/Nombre DNS)
  - Identificación del usuario
- Veremos la versión 2.4 (actual) y la 2.2 (aún en uso)

# Control de acceso (2.2)

- Order Deny,Allow
  - El acceso **se permite** por defecto
  - Si se prohíbe con Deny no hay acceso
  - Si se prohíbe con Deny y se permite con Allow, hay acceso
- Order Allow,Deny
  - El acceso **se prohíbe** por defecto
  - Si se permite con Allow, hay acceso
  - Si se permite con Allow y se prohíbe con Deny, no hay acceso

```
<Directory /var/www/html/daw>  
  Options Indexes  
  AllowOverride None  
  Order Allow,Deny  
  Allow From 127.0.0.0/8  
  Allow From 192.168.3.100  
</Directory>
```



# Ejemplos Deny,Allow

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

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

## Listado 6: Ejemplo de Deny,Allow

```
Order Allow, Deny    # Acceso denegado por defecto
Allow from daw.org   # Se permite el acceso a los hosts de dominio *.daw.org
Deny from bbdd.daw.org # 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
```

## Listado 7: Ejemplo de Allow,Deny

### Ejercicio

¿Qué ocurre si en el segundo ejemplo se utiliza Deny,Allow

# Control de acceso (2.4)

- Por defecto, se deniega todo
- Se permite con Require
- <http://httpd.apache.org/docs/2.4/upgrading.html#run-time>
- [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 (2.4)

## Todo prohibido

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

```
Require all denied
```

## Todo permitido

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

```
Require all granted
```

## Solo se permite a \*.example.org

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

```
Require host example.org
```

# Autenticación y autorización

- **Autenticación:** Verificar la identidad de un usuarios
- **Autorización:** Decidir si un usuario tiene o no permiso para realizar alguna tarea
- Se necesita autenticar a los usuarios para poder aplicar las autorizaciones
  - Hasta ahora, el proceso de autenticación se limitaba a comprobar la dirección IP/DNS del cliente

# Tipos de Autenticación

- **Basic:** Módulo `mod_auth_basic`. Contraseña sin cifrar.
- **Digest:** Módulo `mod_auth_digest`. Contraseña *debilmente* cifrada.
- **Formularios HTML:** Realizada en la aplicación web
- **Certificados digitales:** Módulo `mod_ssl`

# Tipos de autenticación (II)

- Se pueden utilizar módulos adicionales para otros tipos de autenticación.
  - Ficheros de texto
  - Bases de datos
  - Servidores de directorios (LDAP)
  - ...
- `http://httpd.apache.org/docs/2.4/es/howto/auth.html`
- `http://httpd.apache.org/docs/2.4/es/mod/core.html#require`

# Autenticación *Basic*

- 1 Crear fichero con usuarios/contraseñas
- 2 Configurar <Directory> para utilizar el fichero de usuarios/contraseñas
- 3 Configurar <Directory> para requerir usuarios

<http://httpd.apache.org/docs/2.4/es/programs/htpasswd.html>

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

# Nuevo usuario al fichero
htpasswd /etc/apache2/passwd profesor2

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

**Listado 8:** Creación de un fichero de usuarios/contraseñas

# Directivas de autenticación

- AuthType: Basic, Digest, ... (dependiendo de los módulos activados)
- Para Basic
  - AuthName: Nombre del dominio (zona para la que se pide autenticación)
  - AuthUserFile: Creado con htpasswd
  - Require User: Usuarios permitidos

```
<Directory /var/www/html/profesor>  
  Options Indexes  
  AuthType Basic  
  AuthName "Acceso a profesores"  
  AuthUserFile "/etc/apache2/passwd"  
  Require user profesor1 profesor2  
</Directory>
```



**Listado 9:** Ejemplo de autenticación Basic



# Autenticación Digest

<http://httpd.apache.org/docs/2.4/es/programs/htdigest.html>

```
# La primera vez que se invoca el comando se
# utiliza la opcion -c para crear el fichero
htdigest -c /etc/apache2/passwd-digest Informatica admin1

# Nuevo usuario al fichero
htdigest /etc/apache2/passwd-digest Informatica admin2

# Borrar un usuario del fichero
htdigest -D /etc/apache2/passwd-digest Informatica admin1
```

**Listado 10:** Creación de un fichero de usuarios/contraseñas

# Autenticación Digest

```
<Directory /var/www/html/departamento>  
  Options Indexes  
  AuthType Digest  
  AuthName "Informatica"  
  AuthDigestProvider file  
  AuthUserFile "/etc/apache2/passwd-digest"  
  Require user admin1 admin2  
</Directory>
```

**Listado 11:** Ejemplo de autenticación Digest

# Combinaciones de Require

- Las directivas Require tienen un O lógico implícito
- Esto puede cambiarse con RequireAll, RequireAny y RequireNone
  - [http://httpd.apache.org/docs/2.4/mod/mod\\_authz\\_core.html](http://httpd.apache.org/docs/2.4/mod/mod_authz_core.html)

```
<Directory /var/www/html/profesor>
Options Indexes
AuthType Basic
AuthName "Acceso a profesores"
AuthUserFile "/etc/apache2/passwd"
<RequireAll>
    Require user profesor1 profesor2
    <RequireAny>
        Require ip 127.0.0.1
        Require ip 129.168.1.16
    </RequireAny>
</RequireAll>
</Directory>
```

**Listado 12:** Ejemplo de combinación de Require

# Ficheros .htaccess

- Permiten configurar directorios sin modificar los ficheros de configuración de **Apache**
  - Muy útil en hostings donde se comparte una misma instalación de **Apache**
- Se les puede considerar un *include* de un fichero dentro de la directiva Directory

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

**Listado 13:** Fichero de configuración

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

**Listado 14:** Fichero .htaccess en  
/home/profesor/blog

# Ficheros .htaccess

- Cada vez que se produce una petición:
  - Se comprueba la directiva AllowOverride dentro del Directory del recurso
  - El servidor busca en la ruta del recurso que ha solicitado el cliente un fichero con el nombre .htaccess
  - Se aplican sobre el directorio las directivas ya definidas en Directory con las definidas en .htaccess y que han sido permitidas en AllowOverride
- Problemas
  - Rendimiento
  - Seguridad

# Monitorización

- Generalmente, a través de ficheros de registro (*logs*)
- Por defecto, en el directorio `/var/log/apache2`
  - `error.log`: Errores
  - `access.log`: Accesos
- Se pueden utilizar comandos típicos para ficheros de texto
  - `tail -f`, `cat`, `cut`, `awk`, `grep`, `sort`, `uniq...`
- Configurable mediante directivas
  - `ErrorLog`, `LogLevel`, `CustomLog`, `LogFormat...`
  - <http://httpd.apache.org/docs/2.4/en/logs.html>

# Monitorización

## Acceso desde localhost

Ejecuta un comando que muestre por la consola cada acceso que se produzca desde el propio servidor web, pero que no muestre los que se produzcan desde la máquina real.

El comando se quedará en ejecución, mostrando en tiempo real los accesos.

## Recursos accedidos

Ejecuta un comando que muestre por la consola los recursos accedidos, sin duplicados.

# Monitorización

- Existen módulos para analizar los ficheros de traza:
  - mod\_status
  - mod\_info





# Monitorización

## `mod_status`

Habilita el módulo `mod_status` y comprueba su funcionalidad

## `mod_info`

Habilita el módulo `mod_info` y comprueba su funcionalidad

# Otros analizadores de *logs*

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

# Ampliación

- Para saber más
  - <http://www.w3c.org>
  - <http://httpd.apache.org>
  - <http://www.apache.org>
  - [Servicios de Red e Internet \(ISBN: 978-84-1622-832-4\)](#)  
Editorial Garceta.