

Administración de servidores web

Despliegue de aplicaciones web

DAW – Despliegue de aplicaciones web (2º - 0614)

Configuración avanzada del servidor web

- Se distinguen las siguientes categorías:
 - Directivas de control de funcionamiento: permiten controlar el flujo de trabajo del servidor de forma global.
 - Parámetros del servidor web: son parámetros globales para todos los hosts virtuales configurados.
 - Configuración de hosts virtuales: permite crear nombres de dominio diferentes para acceder al servidor desde IP diferentes y puertos diferentes.
 - Instalación y configuración de módulos.
 - Monitorización y rendimiento del servidor: Controlar los accesos al servidor, hosts virtuales, y monitoriza las conexiones y demás problemas o fallos del servidor.

Configuración avanzada del servidor web



Documento
Adobe Acrobat

- Las **Directivas de control del servidor Apache** permiten controlar el funcionamiento del servidor y se localizan en el fichero `apache2.conf`. Véase documento [“Directivas de control del servidor Apache”](#)
- Los **Parámetros del servidor principal** permiten a los hosts virtuales poseer una configuración por defecto:
 - **ServerAdmin**. Este parámetro sirve para suministrar una dirección de email del administrador del sistema al usuario que se conecta mediante un host virtual y será mostrada en las páginas de error generadas. Permitirá ponerse en contacto con el administrador del sistema.
 - **ServerName**. Es el nombre del servidor o una dirección IP válida y un puerto opcional. Esta directiva se puede usar tanto en el fichero principal como en los hosts virtuales. La nomenclatura sería `ServerName domain-name | IP [:port]`.
 - **ServerAlias**. Esta directiva permite establecer nombres adicionales y complementarios al host declarado en la directiva anterior. Su nomenclatura sería `ServerAlias hostname [hostname]...`
 - **DocumentRoot**. Esta directiva establece el directorio a partir del cual es visible la ruta desde el servidor web. La nomenclatura es `DocumentRoot directory-path`.
 - **DirectoryIndex**. Es la lista de recursos que se van a buscar cuando un cliente solicite un directorio. La nomenclatura es `DirectoryIndex disabled | local-url [local-url]...` Permite declarar varios tipos de ficheros como `index.html`, `index.jsp`, `index.htm`, etc.
 - **ErrorDocument**. Es la directiva que permite establecer una página de error para que la visualice el cliente en caso de no poder atender la petición por la causa que sea. La sintaxis sería `ErrorDocument error-code document`.
 - **AccesFileName**. Es la directiva que permite el acceso limitado a determinados directorios.

```
/etc/apache2/  
├─ apache2.conf  
├─ conf-enabled  
├─ magic  
├─ mods-enabled  
├─ sites-available  
├─ conf-available  
├─ envvars  
├─ mods-available  
├─ ports.conf  
└─ sites-enabled
```

```
/etc/apache2/sites-available/  
├─ 000-default.conf  
└─ default-ssl.conf  
  
/var/www/html  
└─ index.html  
  
/etc/apache2/mods-available/mime.conf  
  
/etc/apache2/apache2.conf
```

Host virtuales

- Host virtuales **basados en nombre**. Un servidor web que tiene una sola IP y está IP tiene asociados diferentes dominios.

- En el fichero “/etc/hosts” insertar (para el navegador en local)

```
10.0.2.15 empresa1.com www.empresa1.com
```

```
10.0.2.15 empresa2.com www.empresa2.com
```

- En “sites-enabled” (p.e. en fichero “000-default.conf”), se crean dos servidores virtuales:

```
<VirtualHost empresa1.com:* www.empresa1.com:*>  
    DocumentRoot /var/www/html-empresa1  
    ServerName www.empresa1.com  
    ServerAlias empresa1.com  
</VirtualHost>
```

```
<VirtualHost empresa2.com:* www.empresa2.com:*>  
    DocumentRoot /var/www/html-empresa2  
    ServerName www.empresa2.com  
    ServerAlias empresa2.com  
</VirtualHost>
```

- Otros (opcionales): ServerAdmin, LogFormat, ErrorLog y CustomLog

Host virtuales

- Host virtuales **basados en IP**. Un servidor web que tiene diferentes IP públicas, ya sean físicas o virtuales, y cada dirección web está asociada a una IP.

- Habilitar una segunda IP en el servidor (10.0.3.15), por ejemplo habilitando un segundo adaptador de red en la máquina virtual.
- En el fichero “/etc/hosts” insertar (para el navegador en local)

```
10.0.2.15 empresa1.com www.empresa1.com
```

```
10.0.3.15 empresa2.com www.empresa2.com
```

- En “sites-enabled” (p.e. en fichero “000-default.conf”), se crean dos servidores virtuales:

```
<VirtualHost 10.0.2.15:*>
    DocumentRoot /var/www/html-empresa1
    ServerName www.empresa1.com
    ServerAlias empresa1.com
</VirtualHost>
```

```
<VirtualHost 10.0.3.15:*>
    DocumentRoot /var/www/html-empresa2
    ServerName www.empresa2.com
    ServerAlias empresa2.com
</VirtualHost>
```

- Otros (opcionales): ServerAdmin, LogFormat, ErrorLog y CustomLog

Host virtuales

- Host virtuales **mixtos**. Mezcla del basado en nombre y basado en IP.
 - Habilitar una segunda IP en el servidor (10.0.3.15), por ejemplo habilitando un segundo adaptador de red en la máquina virtual.
 - En el fichero “/etc/hosts” insertar (para el navegador en local)

```
10.0.2.15 empresa1.com www.empresa1.com
10.0.2.15 empresa2.com www.empresa2.com
10.0.3.15 empresa3.com www.empresa3.com
```
 - En “sites-enabled” (p.e. en fichero “000-default.conf”), se crean dos servidores virtuales:

```
<VirtualHost 10.0.2.15:*>
    DocumentRoot /var/www/html-empresa1
    ServerName www.empresa1.com
    ServerAlias empresa1.com
</VirtualHost>

<VirtualHost 10.0.2.15:*>
    DocumentRoot /var/www/html-empresa2
    ServerName www.empresa2.com
    ServerAlias empresa2.com
</VirtualHost>

<VirtualHost 10.0.3.15:*>
    DocumentRoot /var/www/html-empresa3
    ServerName www.empresa3.com
    ServerAlias empresa3.com
</VirtualHost>
```

- Otros (opcionales): ServerAdmin, LogFormat, ErrorLog y CustomLog

Módulos: instalación, configuración y uso

- Para instalar/desinstalar un módulo: `#apt-get install nombre_modulo` `#apt-get remove nombre_modulo`
- Para saber los módulos activos (static o shared): `#apachectl -M`
 - Módulos disponibles con la instalación existente `/etc/apache2/mod-available`
 - Módulos activos (enlaces simbólicos al anterior `/etc/apache2/mod-enabled`
- Existen dos comandos fundamentales para el funcionamiento de los módulos en el servidor web Apache:
 - **#a2enmod**: Utilizado para habilitar un módulo de apache. Sin ningún parámetro preguntará que módulo se desea habilitar. Los ficheros de configuración de los módulos disponibles están en `/etc/apache2/mods-available/` y al habilitarlos se crea un enlace simbólico desde `/etc/apache2/mods-enabled/`.
 - **#a2dismod**: Utilizado para deshabilitar un módulo de Apache. Sin ningún parámetro preguntará que módulo se desea deshabilitar. Los ficheros de configuración de los módulos disponibles están en `/etc/apache2/mods-available/` y al deshabilitarlos se elimina el enlace simbólico desde `/etc/apache2/mods-enabled/`.
 - Si no dispones de esos comandos para poder habilitar y deshabilitar módulos Apache simplemente haces lo que ellos: crear los enlaces simbólicos correspondientes desde `/etc/apache2/mods-enabled/` hasta `/etc/apache2/mods-available/`.
- Módulos de interés:
 - Módulo SSL: permite implementar el protocolo de seguridad SSL.
 - Módulo LDAP: permite la validación con el servicio de directorio.
 - Módulo PHP: se puede ejecutar PHP en el servidor web.
 - Módulo UserDir: permite que todos los usuarios tengan una página web en su directorio personal.
 - Módulo Security: permite bloquear contenidos sobre la base de datos.
 - Módulo Proxy: el servidor Apache se convierte en un proxy inverso.

Autenticación y control de acceso a directorios

- Conceptos:
 - **Control de acceso:** es la acción que permite controlar el acceso a determinados dispositivos a través de su IP o una red de IP.
 - **Autenticación:** es una acción más específica y permite a Apache dar acceso a un recurso mediante la validación de un usuario y contraseña.
- Esquemas/Métodos para autenticar usuarios de Apache:
 - **Método Basic** (mod_auth_basic): este esquema consiste en que el cliente solicita un recurso del servidor y el servidor le solicita un usuario y una contraseña para darle acceso al recurso. Pero el usuario y la contraseña viajan en texto claro (requeriría protocolo SSL).
 - **Método Digest** (mod_auth_digest): permite autenticar un usuario mediante su contraseña pero estos datos son transmitidos cifrados mediante una función hash.
- Limitar recursos a usuarios o grupos en Apache:
 - **Módulo authz_user:** permite dar o denegar a uno o varios usuarios al recurso que se considere.
 - **Módulo authz_group:** permite dar o denegar a uno grupo de usuarios al recurso que se considere.

Autenticación y control de acceso a directorios

- Establecimiento del control de acceso
 - El control de acceso permite controlar el acceso por parte de un dispositivo de la red que desea acceder a un recurso determinado (ficheros y directorios) del servidor Apache.
 - Se aplica dentro de la directiva de <Directory>

Ejemplos

Directiva	Descripción
Require all granted	Se permite el acceso a todo el mundo.
Require all denied	Se deniega el acceso a todo el mundo.
Require env env-var	Se permite el acceso solo a las variables de entorno definidas.
Require method http-method	Acceso a los métodos HTTP definidos.
Require expr expression	Acceso permitido al resultado verdadero de evaluar la expresión definida.
Require user userid	Solamente tienen acceso los usuarios definidos.
Require group group-name	Solamente tienen acceso los grupos definidos.
Require valid-user	Todos los usuarios validados tienen acceso al recurso.
Require ip	Los clientes que tengan como IP o rangos de IP tienen acceso al recurso.

Nadie tiene acceso al directorio /var/www

```
<Directory "/var/www">  
Require all denied  
</Directory>
```

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

Todo el mundo tenga acceso al directorio /var/www

Acceso al directorio /var/www la red 192.168.10.0/24

```
<Directory "/var/www">  
Require all denied  
Require ip 192.168.10.0/24  
</Directory>
```

Autenticación y control de acceso a directorios

- Autenticación básica (módulo adicional AuthGroupFile)
 - Control de acceso en el directorio (fichero .htaccess).

```
/var/www/html-empresa2/delimitado
```

```
AuthType Basic
AuthName "Documentos de Empresa"
AuthUserFile "/etc/apache2/web.htpasswd"
Require user admin
```

- Creación de usuarios ("admin")

```
#htpasswd -c /etc/apache2/web.htpasswd admin
```

- Directiva <Directory> en el site

```
<Directory "/var/www/html-empresa2/delimitado/">
    AllowOverride AuthConfig
    Options Indexes FollowSymLinks MultiViews
    Order allow,deny
    Allow from all
</Directory>
```

Autenticación y control de acceso a directorios

- Autenticación mediante LDAP (módulo authnz_ldap)
 - Control de acceso en el directorio (fichero .htaccess).

```
/var/www/html-empresa2/delimitado
```

```
AuthName "Autenticacion por LDAP"  
AuthType Basic  
AuthBasicProvider ldap  
AuthzLDAPAuthoritative on  
AuthLDAPUrl ldap://127.0.0.1/ou=usuarios,dc=proyecto,dc=com?uid  
Require ldap-user user1LDAP
```

Certificados

- Servidor virtual seguro en apache (módulo mod_ssl) `#a2enmod ssl`
 - Creación del directorio donde se almacenaran los certificados
`#mkdir /etc/apache2/empresa #mkdir /etc/apache2/empresa/ssl`
 - Creación del certificado y clave privada (en directorio /etc/apache2/empresa/ssl)
`#openssl req -new -nodes -keyout empresa.key -out empresa.csr`
 - Autofirma del certificado (como AC)
`#openssl x509 -in empresa.csr -out empresa.crt -req -signkey empresa.key -days 3650`
 - Asociarlo a un site (puerto 443)

```
<VirtualHost 10.0.3.15:443>

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html-empresa3
    ServerName www.empresa3.com
    ServerAlias empresa3.com

    SSLEngine on
    SSLCertificateFile /etc/apache2/empresa/ssl/empresa.crt
    SSLCertificateKeyFile /etc/apache2/empresa/ssl/empresa.key

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

</VirtualHost>
```

Pruebas de funcionamiento, monitorización y rendimiento del servidor web

- Registro y monitorización
 - En el fichero de configuración `apache2.conf` existen dos registros principales para controlar el servidor:
 - Registro de accesos: se almacenan las peticiones solicitadas al servidor. De cada cliente registra la IP, la fecha y hora de la petición, el tipo de petición (GET, POST), etc.

`/var/log/apache2/access.log`

- Registro de errores: se almacenan los errores provocados por una IP o por un error en la configuración del servidor y que provoca que el servicio no pueda levantarse

`/var/log/apache2/error.log`

Campos de ficheros de log

Campos	Descripción	Ejemplo de log
host(%h)	Indica la dirección IP del cliente que hizo la petición al servidor. Si está activada la directiva HostnameLookups, el servidor mostrará el nombre del host.	192.168.1.20
ident(%l)	Esta información está relacionada con la identidad de la máquina del cliente. Esta información es poco fiable, por lo que casi nunca se usa. El guion significa que la información no está disponible.	-
authuser(%u)	Identifica el usuario que solicita el recurso mediante la autenticación HTTP.	-
date(%t)	Muestra la fecha y hora en la que se ha realizado la petición del recurso al servidor.	[08/Jul/2020:05:38:30-0400]
request(%r)	Indica la petición del cliente. Consta del método usado que es GET, el protocolo usado que es HTTP/1.1.	"GET / HTTP/1.1"
status(%s)	Es el código de estado que devuelve el servidor al cliente.	403
bytes(%b)	El número de bytes devueltos al cliente.	497

Pruebas de funcionamiento, monitorización y rendimiento del servidor web

Directivas	Descripción
LogFormat	Directiva que define el formato que tendrá el fichero de log. En esta directiva se podrán indicar los campos parametrizables detallados en el cuadro 4.2. Este formato de log será usado en la directiva TransferLog y CustomLog para poder customizar el fichero de log como se adapte a nuestras necesidades.
CustomLog	Directiva que permite almacenar en el fichero de log las solicitudes de acceso al servidor Apache. Esto es, cada petición HTTP será almacenada en este fichero de log, y además se permite personalizar los campos que pueda contener cada línea almacenada en el fichero de log.
CookieLog	Directiva que permite almacenar información sobre las cookies en un fichero. La ruta del fichero partirá del directorio que se configure en la directiva ServerRoot.
ErrorLog	Indica el nombre del fichero donde se almacenarán los mensajes de error generados por el servidor Apache. Se puede usar syslog en lugar del nombre del fichero siempre que el sistema lo soporte.
RewriteLog	Directiva que permite almacenar en un fichero los logs del servidor reescribiendo acciones realizadas.
ScriptLog	Establece un fichero para registrar los errores de script CGI. Si no se especifica esta directiva, no se creará ningún fichero de errores.
TransferLog	Directiva similar a la directiva CustomLog, con la excepción de que no permite especificar explícitamente el formato de log. Toma por defecto el formato definido en la directiva LogFormat.

Directivas de logs

Pruebas de funcionamiento, monitorización y rendimiento del servidor web

- Pruebas de rendimiento

- Instalación (Apache Bench)

`#apt-get install apache2-utils`

- Ejecución (1500 peticiones y 25 usuarios)

`#ab -n 1500 -c 25 https://www.empresa3.com/`

```
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking www.empresa3.com (be patient)


Server Software:      Apache/2.4.41
Server Hostname:      www.empresa3.com
Server Port:          443
SSL/TLS Protocol:     TLSv1.2,ECDHE-RSA-AES256-GCM-SHA384,2048,256
Server Temp Key:      X25519 253 bits
TLS Server Name:      www.empresa3.com

Document Path:        /
Document Length:      10910 bytes

Concurrency Level:     25
Time taken for tests:  4.127 seconds
Complete requests:     1500
Failed requests:       0
Total transferred:     16776000 bytes
HTML transferred:     16365000 bytes
Requests per second:   363.49 [#/sec] (mean)
Time per request:      68.777 [ms] (mean)
Time per request:      2.751 [ms] (mean, across all concurrent requests)
Transfer rate:         3970.00 [Kbytes/sec] received

Connection Times (ms)
              min    mean[+/-sd] median    max
Connect:        2     58  11.2      56     175
Processing:      1     10   6.7       9     163
Waiting:         0      6   5.7       6     163
Total:          17     68  11.9      65     212

Percentage of the requests served within a certain time (ms)
 50%    65
 66%    69
 75%    72
 80%    75
 90%    82
 95%    88
 98%    95
 99%   109
100%   212 (longest request)
```