TEMA 2
Servidores Web

Contenidos

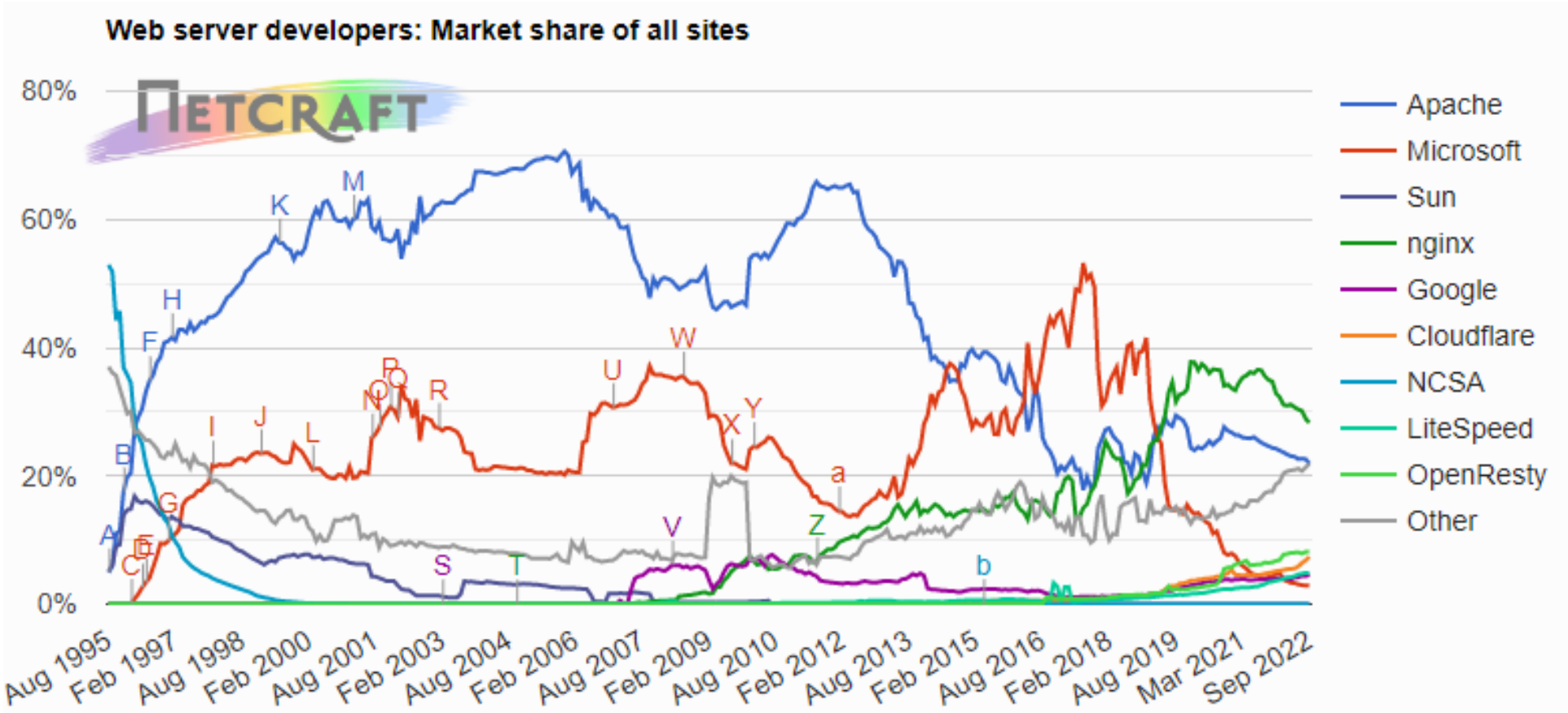
Características de Apache

Apache Web Server es uno de los principales proyecto de la comunidad Open Source Apache. Ha sido tradicionalmente el servidor web más utilizado, y su versión más actual es la 2.4.5

Es altamente escalable, configurable, editando ficheros de texto, y extensible, a través de módulos que se pueden crear con código (Pearl, C)

Apache soporta varios “sites” lógicos, basados en IP o nombres de dominio, sobre la misma instalación física.

Cuota de Mercado de Web Servers



Ficheros de Configuración

Dentro del directorio /etc/apache2/ hay varios archivos de configuración

apache2.conf

- Fichero principal de configuración
- Contiene varias directivas
- Incluye otros ficheros de configuración (include)

ports.conf

- Se definen las ips y puertos donde escucha cada servidor (virtual)
- Incluido en apache2.conf

envar

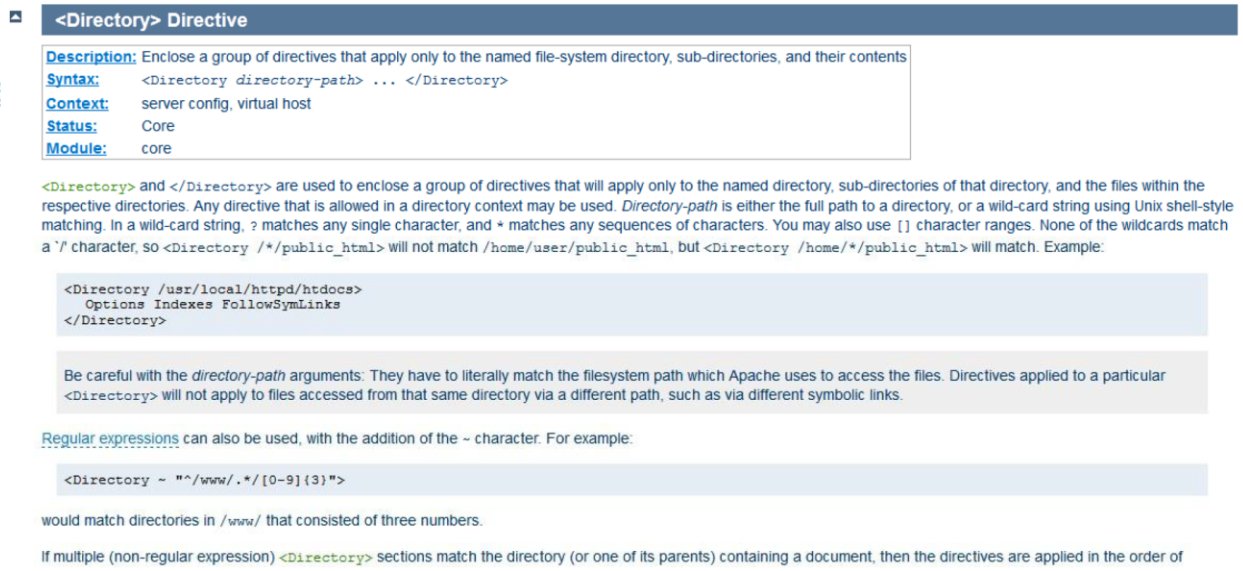
- Variables de entorno

```
# 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/
```

Directivas

Están dentro de los ficheros de configuración. Las que no se especifican toman sus valores por defecto. Hay un gran número de ellas.

- DefaultIcon
- DefaultLanguage
- DefaultType
- DeflateBufferSize
- DeflateCompressionLe
- 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

Referencias

Directorios de Configuración

Dentro del directorio /etc/apache2/ hay varios directorios en parejas

/conf-available/	• Configuraciones disponibles (no todas activas)
/conf-enabled/	• Configuraciones Activas • Enlaces a los archivos del directorio anterior
/modules-available/	• Módulos disponibles (no todas activas)
/modules-enabled/	• Modulos Activas (habilitados) • Enlaces a los archivos del directorio anterior
/sites-available/	• Configuración del site virtual por defecto • Se pueden añadir más
/sites-enabled/	• Sites Activos (habilitados) • Enlaces a los archivos del directorio anterior

Comandos Útiles

Cada vez que se cambia un fichero de configuración es necesario parar y reiniciar el servicio para que los cambios tengan efecto, o solicitar una recarga

Parada	$ sudo service apache2 stop
Inicio	$ sudo service apache2 start
Reinicio	$ sudo service apache2 restart
Recarga	$ sudo service apache2 reload
Status	$ sudo service apache2 status
Control	$ apachectl (–M, -c)

Contenidos

1. **Introducción**
   - Características de Apache
   - Parada y Reinicio del Servicio
   - Configuración
2. **Módulos, Servicios Virtuales y Monitorización**
   - Módulos
   - Servicios Virtuales
   - Monitorización
3. **Acceso y Verificación**
   - Control de Acceso
   - Autenticación y Autorización
   - Certificados

Módulos

Los módulos extienden la funcionalidad de servidor. Cada módulo tiene una funcionalidad y una configuración propia (.conf)

Tipos	• Estáticos: parte del software (compilados). Más rápidos • Dinámicos: al iniciar el servidor (directivas, comandos). Más flexible
Directivas	• LoadModule: permite cargar un modulo • <IfModule> *Nombre del modulo* </IfModule>: acciones si el modulo está cargado
Disponibles/ Habilitados	• /etc/apache2/mods-available/ • /etc/apache2/mods-enabled/ -> commando a2enmod/a2dismod *Módulo*
Comandos	• $ a2enmod *Módulo* : Habilita el módulo (crea la entrada en mods-enabled • $ a2dismod *Módulo* : Deshabilita el módulo (borra la entrada)

Módulos Habituales

Algunos módulos de interés

SSL	• Permite la implantación del protocolo de seguridad SSL
LDAP	• Validación con el servicio de directorio
PHP	• Ejecución de PHP en el servidor
UserDir	• Permite que todos los usuarios tengan una página web en su directorio
Security	• Permite bloquear contenidos sobre la base de datos
Proxy	• El servidor Apache se convierte en un proxy inverso

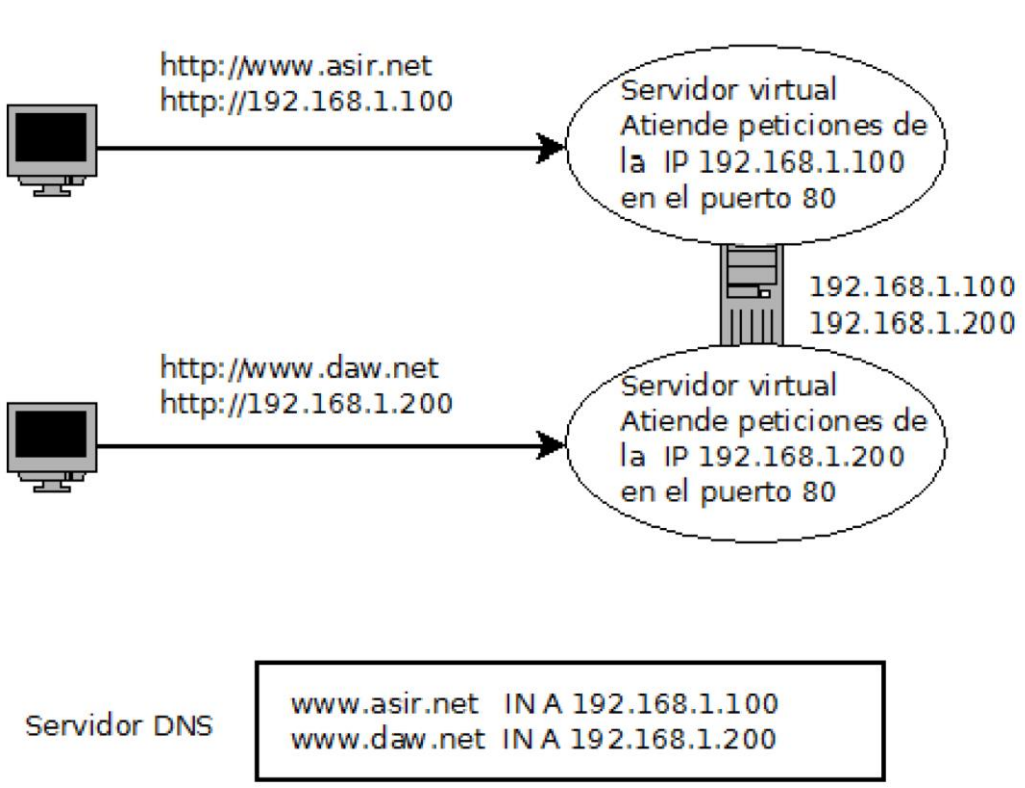
Servidores Virtuales

Permiten “simular” varios servidores web en la misma máquina física. Su objetivo es maximizar el uso de recursos: CPU, Memoria.. y direcciones IP

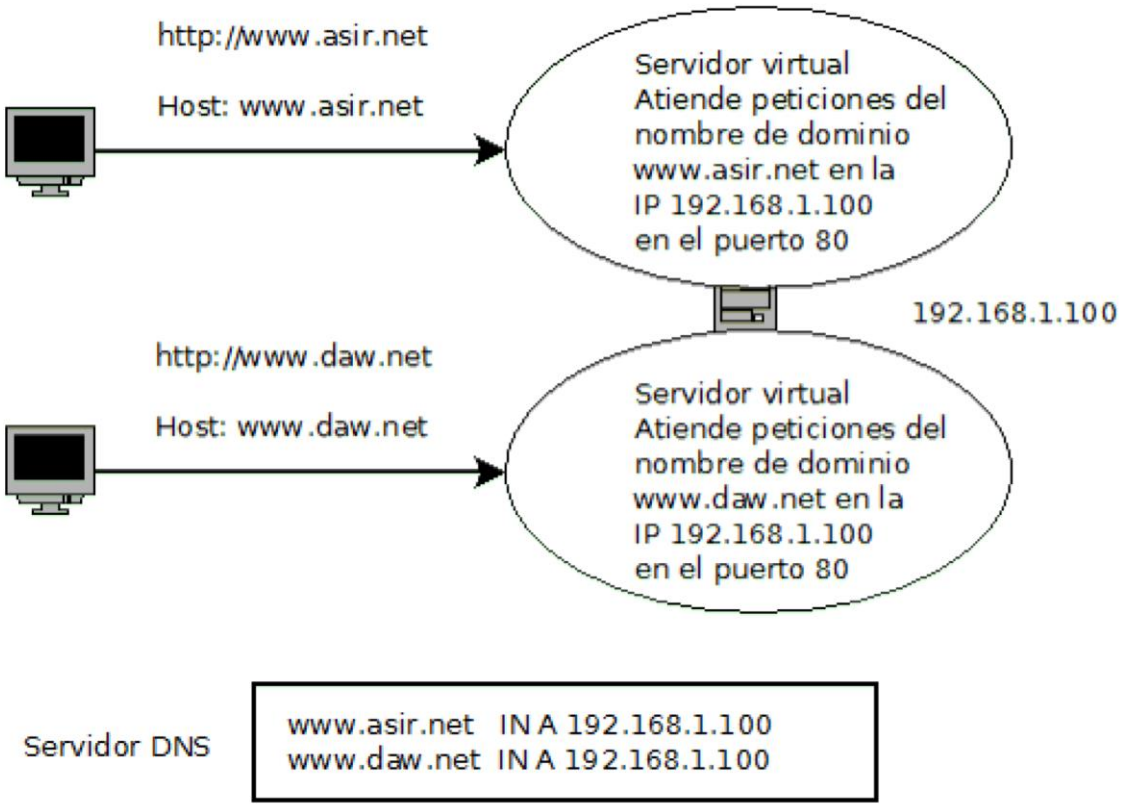
Basado en IPs	• Cada servidor virtual tiene una IP diferente
Basado en Nombres	• Todos los servidores comparten la misma IP, y las respuestas se controlan por el nombre del dominio de la petición
Servidores DNS	• Es necesario configurar los servidores DNS de acuerdo a la modalidad (nombre o IP)

Servidores Virtuales

BASADO EN IP



BASADO EN NOMBES



Monitorización

Es una de las tareas fundamentales de la administración de un servidor Apache, revisando los logs. El formato de los logs es parametrizable

Ficheros de Log	• Accesos -> /var/log/apache2/access.log • Errores -> /var/log/apache2/error.log
Directivas	• Los ficheros de log se pueden controlar con varias directivas directivas: ErrorLog, LogLevel, LogFormat, CustomLog... • Lo más relevante de los logs es el formato y el nivel
Módulos	• mod_status -> Rendimiento del servidor (recursos) • mod_info -> Configuración
Analizadores	• Webalizer • Awstats

Contenidos

1. **Introducción**
   - Características de Apache
   - Parada y Reinicio del Servicio
   - Configuración
2. **Módulos y Servicios Virtuales y Monitorización**
   - Módulos
   - Servidores Virtuales
   - Monitorización
3. **Acceso y Verificación**
   - Control de Acceso
   - Autenticación y Autorización
   - Ficheros .htaccess

Control de Acceso

Se establecen directivas dentro de la etiqueta <Directory>:

Options:

- Indexes: Si no hay un fichero index.html, Apache devuelve la lista de los ficheros html del directorio
- FollowSymLinks: Permite seguir enlaces simbólicos (rendimiento)

AllowOverride:

- All: Permite sobreescribir las directivas con ficheros .htaccess en cada directorio
- None: no hay uso de .htaccess

<Directory /var/www>
  Options Indexes, FollowSymLinks
  AllowOverride *None, All*
  Require all granted
</Directory>

Control de Acceso - Require

Apache determina si una identidad tiene acceso a un directorio mediante la directiva **Require** dentro de <Directory>

all granted	• Se permite el acceso a todo el mundo
all denied	• Se deniega el acceso a todo el mundo
ip	• Permitidos esas IPs o Rangos
user *userid*	• Solamente tienen acceso los usuarios definidos
group *group-name*	• Acceso al grupo
valid-user	• Todos los usuarios validados pueden acceder
env *env-var*	• Se permite el acceso a las variables de entorno definidas
method *http-method*	• Acceso a los métodos http definidos
expr *expression*	• Resultado verdadero de evaluar la expresión definida

Fichero .htaccess

- Ficheros que permiten la configuración personalizada de directorios.
- Contienen las mismas directivas que el fichero general apache2.conf pero sólo aplican al directorio donde se encuentran
- 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 y aplica sobre el directorio las directivas definidas.
- En la configuración del servidor hay que permitir el uso de estos ficheros.
- No se deben usar a menos que no se tenga acceso al archivo de configuración del servidor (Ej.: Servidor de hosting)
- El nombre .htaccess se puede cambiar con la directiva AccessFileName.

Autenticación - Tipos

Consiste en validar que una identidad es quien dice ser solicitando una contraseña que se contrasta con la contraseña correcta que se guarda en un fichero.

Básica	• Módulo: mod_aut_basic • La contraseña viaja "en claro" • Las contraseñas se almacenan en el fichero /etc/apache2/password
Digest	• Módulo: mod_aut_digest • La contraseña viaja cifrada (hash) • Las contraseñas se almacenan en el fichero /etc/apache2/digest
LDAP	• Módulos que ofrecen la posibilidad de acceder a credenciales (usuarios, contreras, certificados, …), especialmente **LDAP**

Autenticación - Basic

Para realizar la autenticación Basic hay que seguir los siguientes pasos:

1. Verificar que el módulo mod_aut_basic está instalado
2. Instalar el paquete apache2-utils

   sudo apt –get install apache2 –utils
3. Crear el fichero passwd

   $ sudo htpasswd –c /etc/apache2/passwd
4. Añadir los usuarios al fichero

   $ sudo htpasswd /etc/apache2/passwd *usuario*

Autenticación - Basic

5. Añadir las directivas en apache2.conf o .htaccess

```
<Directory /var/www/html/directorio>
        Options Indexes FollowSymLinks
        AllowOverride None
        AuthType Basic
        AuthName “Acceso restringido”
        AuthUserFile /etc/apache2/passwd
</Directory>
```

Autenticación - Digest

Para realizar la autenticación Digest hay que seguir los siguientes pasos:

1. Verificar que el módulo mod_aut_digest está instalado
2. Instalar el paquete apache2-utils

```
sudo apt –get install apache2 –utils
```

3. Crear el fichero passwd

```
$ sudo htdigest –c /etc/apache2/passwd
```

4. Añadir los usuarios al fichero

```
$ sudo htdigest /etc/apache2/passwd usuario
```

Autenticación - Digest

5. Añadir las directivas en apache2.conf o .htaccess

```
<Directory /var/www/html/directorio>
        Options Indexes FollowSymLinks
        AllowOverride None
        AuthType Digest
        AuthName “Acceso restringido”
        AuthUserFile /etc/apache2/passwd
</Directory>
```

Certificados

Para poder establecer una comunicación SSL es necesario disponer en el servidor de un certificado para cifrar la clave de sesión. Ese certificado debe incluirse en el fichero de configuración del host virtual