

Curso online: **Instalación, Configuración y Administración de Apache + Tomcat**

Módulo 2. Apache Web Server

Capítulo 3. Características de Apache Web Server

Autores

Janine García Morera

Alexandra López de la Oliva Portugués

Julio Villena Román

Octubre de 2014

Índice de contenidos

Capítulo 3	Características de Apache Web Server	2
3.1.	Introducción	2
3.2.	Objetivos	2
3.3.	Arquitectura	2
3.3.1.	Apache 1.3	3
3.3.2.	Apache 2.x	4
3.4.	Seguridad	5
3.4.1.	Autenticación	5
3.4.2.	Control de accesos	6
3.4.3.	SSL / TLS	6
3.5.	Rendimiento y escalabilidad	6
3.5.1.	Balanceo de carga	7
3.5.2.	Compresión	7
3.6.	CGI Scripts	8
3.6.1.	Perl	8
3.6.2.	PHP	8
3.6.3.	Otros	8
3.7.	Módulos Apache	9
3.8.	Conclusiones	9

CAPÍTULO 3 CARACTERÍSTICAS DE APACHE WEB SERVER

3.1. Introducción

El **servidor Apache se estructura en módulos**. Una de las principales tareas a la hora de configurar Apache consiste en decidir y personalizar los módulos que se van a utilizar.

Hay tres tipos principales de módulos: los **módulos base**, que soportan las funciones básicas de Apache, los **módulos multiproceso**, íntimamente relacionados con el Sistema Operativo, y los **módulos adicionales**, que no son absolutamente necesarios pero potencian grandemente la funcionalidad de Apache.

En la actualidad hay tres versiones de Apache funcionando: la versión 2.0, la 2.2 y la 2.4, aunque de ellas la versión 2.0 ya no tiene mantenimiento.

Anteriormente existió la versión 1.3 que es la más conocida y la que supuso la gran expansión del servidor.

Conocer los módulos que pueden usarse con Apache permite tener una visión más real de toda la potencialidad que ofrece el que, hoy por hoy, es el servidor Web más extendido y usado de Internet.

3.2. Objetivos

- Conocer la arquitectura del servidor Apache.
- Conocer las mejoras de la versión 2 con respecto a la 1.3.
- Saber cómo puede implementar Apache la seguridad.
- Conocer las posibilidades de mejora del rendimiento de Apache.
- Conocer los módulos disponibles para ampliar la funcionalidad de Apache.
- Saber algo sobre los distintos lenguajes específicos para diseñar scripts CGI, algunos de los cuales son también desarrollados por la ASF.

3.3. Arquitectura

El servidor **Apache está estructurado en módulos**. Cada uno de los módulos se configura mediante las directivas que están contenidas en el propio módulo.

Pueden clasificarse los módulos apache como:

- **Módulos Base:** Módulo con las funciones básicas del Apache.
- **Módulos Multiproceso:** son los responsables de la unión con los puertos de la máquina. Aceptan las peticiones y levantan hilos que las atienden.
- **Módulos Adicionales:** Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiproceso para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código.

El resto de funcionalidades del servidor se consiguen por medio de módulos adicionales que se pueden cargar. Para incrementar las funcionalidades del servidor, simplemente hay que añadirle un módulo, de forma que no es necesario volver a instalar el software. Hay módulos adicionales de código abierto y uso gratuito y módulos comerciales.

Apache es un sistema modular, en el que el administrador puede ampliar la funcionalidad del servidor cargando y seleccionando un nuevo conjunto de módulos, que pueden ser compilados de forma estática en el ejecutable Apache (`httpd`) o pueden ser compilados como Objetos Compartidos Dinámicamente (Dynamic Shared Objects, DSO) que no están integrados en el cuerpo del ejecutable, sino que tienen la forma de librerías dinámicas que se pueden cargar o no, en función de las necesidades.

En la actualidad están funcionando las versiones 2.x de Apache, pero anteriormente había otra versión en funcionamiento la 1.3. Entre la versión 1.3 y las siguientes hay varias diferencias tanto en arquitectura como en capacidades.

3.3.1. Apache 1.3

Apache 1.3 se ha implementado en una gran cantidad de sistemas Unix y Windows. Algunas de sus características más representativas son las siguientes:

- Web server basado en procesos. **Apache 1.3 sobre Unix es un Web Server basado en procesos.** El proceso padre de Apache crea varios “procesos hijos” al arrancar. Cada uno de estos “hijos” puede servir a una petición independiente de los demás. Este enfoque tiene la ventaja de la estabilidad: si uno de los hijos funciona mal (se pierde el control o desborda la memoria, por ejemplo), se le puede matar sin necesidad de que los otros se vean afectados. Sin embargo, esto afecta al rendimiento: en muchos sistemas Unix, la creación de procesos y la asignación de tiempo de procesador a cada uno de ellos son operaciones muy pesadas. Además, puesto que los procesos están aislados los unos de los otros, no pueden compartir código ni datos, por lo que son grandes consumidores de recursos.

- Soporte Windows. **Apache 1.3 es la primera versión de Apache que soporta Windows**, aunque no de forma tan estable como sobre Unix, al haber sido desarrollado sobre Unix y posteriormente portado.
- Modular. **La arquitectura de Apache 1.3 es modular**. Se pueden activar y desactivar módulos para añadir o retirar funcionalidad al Web Server. Existe una base muy importante de módulos nativos, así como de terceros.

3.3.2. Apache 2.x

Apache 2.x es la última rama del servidor Apache. La arquitectura de la versión 2.x presenta algunas mejoras significativas con respecto a la versión 1.3. Algunas de ellas son las siguientes:

- **Módulos de multiproceso**. Apache 2.x abstrae la arquitectura de procesamiento de peticiones en módulos especiales, llamados Módulos de Multiproceso (Multi Processing Modules, MPMs). Esto significa que Apache puede ser configurado como un servidor basado en procesos puro, un servidor multitenhebrado puro o una mezcla de ambos modelos. Los hilos (threads) están contenidos en procesos y se ejecutan simultáneamente. Al contrario que los procesos, los threads pueden compartir datos y código, por lo que son más “ligeros” que los procesos, pero en contrapartida el mal funcionamiento de un thread puede afectar a otros que comparten su código o sus datos.

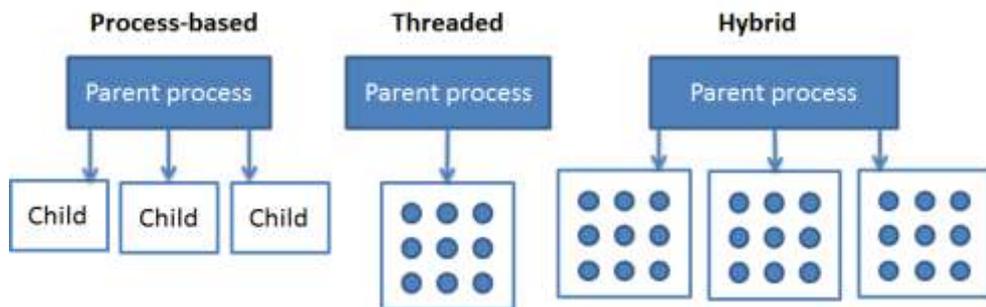


Figura 2.3.1: Esquema de las diferentes configuraciones que puede tomar Apache

- **Módulos de protocolo**. La gestión de protocolos ha sido encapsulado en su propio nivel en la versión 2.x de Apache, lo que significa que se pueden escribir módulos para dar servicio a protocolos diferentes a HTTP. Esto significa que se podrían identificar usuarios POP3, por ejemplo, usando la misma base de usuarios de Apache, o que se podría generar dinámicamente contenido accesible mediante FTP utilizando PHP o CGIs.
- **Arquitectura modular y filtros**. Apache 2.x mantiene la arquitectura modular de la versión 1.3 y añade un nuevo mecanismo: los filtros. Esto permite a los módulos modificar el contenido generado por otros módulos: por ejemplo, se puede escanear, chequear en busca de virus o comprimir no sólo ficheros estáticos, sino contenido generado dinámicamente.
- **Compatibilidad entre versiones**. Los módulos de la versión 1.3 deben ser portados a la versión 2.x. Algunos de los módulos principales, como PHP y mod_perl, tienen sus propias versiones y otros, como mod_dav y mod_ssl, forman parte en la nueva versión de la distribución del servidor. Algunos módulos

de terceros sólo funcionan correctamente en la versión 2.x si Apache se ejecuta como un servidor puro basado en procesos.

- **Portabilidad.** Apache funciona ahora de la misma forma en Unix como en Windows gracias a la librería Apache Portable Runtime (APR) <<https://apr.apache.org/>>. Esta librería abstrae a Apache de las diferencias entre sistemas operativos, tales como las APIs de acceso a ficheros o a red. De esta forma, para portar Apache a una nueva plataforma basta con portar la APR. Además se desarrollan Módulos de Multiprocesamiento (MPMs) específico de cada plataforma, por lo que estas plataformas tienen implementada su propia API nativa, evitando las capas de emulación POSIX que provocan problemas y un bajo rendimiento.
- **Soporte de Ipv6.** En los sistemas que soportan IPv6 con la librería Apache Portable Runtime.

Conceptos importantes	
Mejoras de Apache 2.x respecto a Apache 1.3	Módulos de multiproceso
	Módulos de protocolo
	Arquitectura modular y filtros
	Compatibilidad entre versiones
	Portabilidad
	Soporte de Ipv6

3.4. Seguridad

Apache está provisto de varios módulos relacionados con la seguridad para securizar y restringir el acceso al servidor.

3.4.1. Autenticación

Los módulos de autenticación permiten determinar la identidad de un cliente, normalmente verificando un nombre de usuario y una clave contra una base de datos. Apache incluye módulos para identificar contra un fichero plano y contra ficheros de bases de datos. También existen módulos para conectar Apache a entornos de seguridad y bases de datos, incluyendo NT Domain Controller, Oracle, MySQL, etc.

De especial interés son los **módulos LDAP**, que permiten la integración con servicios de directorio existentes en la organización.

3.4.2. Control de accesos

El módulo `mod_authz_host` o `mod_access_compat` (en las versiones 1.3 y 2.0 se llamaba `mod_access`) permite restringir el acceso a recursos en base a parámetros de la request del cliente, como puede ser la presencia de una cabecera específica o la dirección IP o el hostname del cliente.

Además de ellos, el `mod_authz_core` permite que a los usuarios autenticados se les pueda permitir o denegar el acceso a diferentes partes de la web.

3.4.3. SSL / TLS

Los **protocolos SSL/TLS** (Secure Sockets Layer/Transport Layer Security Protocols) permiten encriptar los datos entre el servidor web y el cliente. En Apache 1.3 estos protocolos se implementaban por medio del módulo `mod_ssl`, distribuido desde el sitio web de `mod_ssl` (www.modssl.org/) y que requería parchear el servidor. A partir de la versión 2.0, `mod_ssl` se incluye como un módulo base con Apache.

Conceptos importantes	
Módulos de Apache relacionados con la seguridad	Autenticación: módulos de autenticación de usuarios contra ficheros planos y contra BB.DD y módulos LDAP
	Control de acceso: <code>mod_access</code> (v1.3 y v2.0) y <code>mod_authz_host</code> (v2.2)
	SSL / TLS: <code>mod_ssl</code>

3.5. Rendimiento y escalabilidad

Hay varios factores a la hora de valorar las características de un servidor Web. Los más importantes son la flexibilidad, la estabilidad y el rendimiento.

Ya se ha indicado anteriormente que Apache es uno de los servidores Web más estables, robustos y flexibles. Además, se utiliza en algunos de los sitios más visitados de Internet (se han dado antes algunos ejemplos), sin notificación de problemas de rendimiento. En cualquier caso, hay formas de mejorar el rendimiento de servidores Web Apache con una gran carga, sobre todo si sirven fundamentalmente contenido estático. De la misma manera se puede medir y controlar el uso de ancho de banda. Algunos de los **módulos que permiten mejorar el rendimiento** son los siguientes:

- **`mod_file_cache`:** se incluye a partir de la versión 2.0, y mapea a memoria una lista estática de ficheros que son accedidos muy frecuentemente pero que rara vez son modificados. Esta funcionalidad también estaba disponible en la versión 1.3 pero en un módulo a parte llamado `mod_nmap`.

- **mod_bandwidth** (<www.cohprog.com/mod_bandwidth.html>): este módulo de Apache 1.3 permite poner límites de ancho de banda a nivel de servidor o por conexión, basándose en un directorio específico, el tamaño de los ficheros o una dirección IP o dominio.
- **bandwith_share_module** (<www.topology.org/src/bwshare/README.html>): permite regular el ancho de banda y balancear por dirección IP del cliente. Está soportado por Apache 1.3 y las primeras releases de la versión 2.0.
- **mod_throttle**: regula el ancho de banda por host virtual o por usuario.
- **mod_ratelimit**: limita el ancho de banda del cliente.
- **mod_bw** (<bwmmod.sourceforge.net/>): permite regular el ancho de banda por host virtual.
- **mod_cband** (<dembol.org/blog/mod_cband/>): permite regular el ancho de banda por host virtual o por usuario.

3.5.1. Balanceo de carga

Por medio de la **funcionalidad de reverse proxy** de Apache y del módulo **mod_rewrite** se puede hacer que un proceso de Apache distribuya las requests entre varios servidores web de back-end. Existen módulos de terceros (como por ejemplo **mod_backhand**, <www.backhand.org>) que permiten realizar balanceo de carga en base a la ocupación de recursos de los servidores.

En la versión 2.1 aparece el módulo **mod_proxy_balancer** que permite realizar balanceo de carga de servicios basados en protocolos HTTP, FTP y AJP13.

3.5.2. Compresión

La versión 2.0 de Apache incluye el módulo **mod_deflate**, que es un módulo de filtrado que comprime el contenido antes de ser remitido al cliente, con lo que se ahorra ancho de banda en detrimento del rendimiento. Esta funcionalidad la realizaba el módulo **mod_gzip** en la versión 1.3.

Conceptos importantes	
Mejora de rendimiento:	mod_map (v1.3), mod_file_cache (v2.0), mod_bandwidth (v1.3), bandwith_share_module (v1.3), mod_throttle (v1.3), ...
Balanceo de carga:	mod_proxy_balancer (v2.1) y mod_rewrite (v1.3)
Compresión:	mod_deflate (v2.0) y mod_gzip (v1.3)

3.6. CGI Scripts

CGI es el acrónimo de **Common Gateway Interface**. Los programas CGI son programas externos que son llamados cuando un usuario solicita una determinada página. El programa CGI puede recibir información desde el servidor Web (valores de variable de un formulario, tipo de navegador, dirección IP del cliente, etc.) y utiliza dicha información para generar y enviar una página web al cliente. En general **un programa CGI puede estar escrito en cualquier lenguaje ejecutable por el servidor** (desde C a scripts Unix o lenguaje de comandos en Windows). Sin embargo hay una serie de módulos basados en lenguajes diseñados casi especialmente para ser utilizados en CGIs.

3.6.1. Perl

`mod_perl` (perl.apache.org) es uno de los proyectos más antiguos de Apache. Embebe un intérprete Perl en Apache, por lo cual se pueden escribir módulos enteros en Perl. El hecho de embeber un intérprete Perl en el servidor Web evita la sobrecarga de lanzar un intérprete externo, por lo que se consigue una ejecución mucho más rápida.

En la versión 1.3 de Apache cada proceso debía tener embebido el intérprete. En las versiones multitenhebradas de Apache v 2.0, `mod_perl` permite compartir código, datos y estado de la sesión entre los intérpretes.

3.6.2. PHP

PHP es un lenguaje interpretado, embebido en HTML, ejecutable en el servidor y portable entre plataformas. Es el módulo más popular de Apache, según fuentes de SecuritySpace (www.securityspace.com). Las razones para esa popularidad son, entre otras:

- Una curva de aprendizaje bastante pequeña.
- Gran cantidad de documentación y de usuarios.
- Amplio soporte de acceso a bases de datos.
- Modularidad.

3.6.3. Otros

Hay otros módulos para el desarrollo de CGIs, menos populares pero también extensamente utilizados: `mod_python`, (www.modpython.org) permite integrar Python, otro popular lenguaje orientado a objetos, con el servidor Web Apache. También existen módulos para integrar Tcl. Una mención especial, puesto que es el objetivo de este curso, es el uso de los módulos `mod_jk` y `mod_proxy_ajp` que permiten la conexión de Apache con Tomcat para ejecutar JavaServer Pages o servlets java.

3.7. Módulos Apache

Se puede ver información detallada de los módulos incluidos en cada distribución en la documentación de Apache. La versión en la que se basa el curso es en la más actual que es la 2.4 <<http://httpd.apache.org/docs/current/en/mod/>>

3.8. Conclusiones

- La arquitectura de Apache se basa en módulos. Las funciones básicas se incluyen en los módulos base.
- Se puede ampliar la funcionalidad de Apache utilizando el soporte DSO.
- La versión 1.3 estaba basada en procesos. Las versiones 2.x puede mantener esta arquitectura o utilizar una basada en hilos o una híbrida.
- Las versiones 2.x han incrementado su portabilidad al introducir el Apache Portable Runtime (APR).
- Apache basa la seguridad en la autenticación de los usuarios, el control de acceso de los clientes y en el cifrado.
- Apache es uno de los servidores Web más estables, robustos y flexibles. Ofrece módulos para poder controlar la carga y mejorar su rendimiento. También permite balancear la carga entre varios servidores y comprimir el contenido antes de mandárselo al cliente para ahorrar ancho de banda.
- Permite ofrecer contenido dinámico incluyendo módulos CGI.