APACHE (Prefork, configuración en httpd.conf)

**Multiprocesamiento (MPM) de Apache prefork**

Un servidor web pre-forking y no hebrado trata las peticiones de una manera similar a como lo hacía Apache 1.3. Esto es apropiado para sitios web que necesitan evitar el hebrado para ser compatibles con librerías que no son seguras cuado se usan hebras. Es también el mejor MPM para aislar cada petición, de manera que si suge un problema con una petición, esto no afecte al resto.

Este MPM está muy autorregulado, de manera que muy pocas veces es necesario ajustar los valores de sus directivas de configuración. El valor que se fije en la directiva [MaxClients](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxclients) debe ser lo suficientemente grande para tratar tantas peticiones simultáneas como espere recibir su sitio web, pero lo suficientemente pequeño para asegurarse de que hay memoria RAM suficiente para todos los procesos.

**¿CÓMO FUNCIONA?**

Un solo proceso de control es el responsable de lanzar los procesos hijo que escuchan las peticiones que se puedan producir y las sirven cuando llegan. Apache siempre intenta mantener varios procesos de sobra o en espera, que estén disponibles para servir peticiones cuando lleguen. Así, los clientes no tienen que esperar a que un nuevo proceso hijo sea creado para ser atendidos.

Las directivas [StartServers](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#startservers), [MinSpareServers](http://httpd.apache.org/docs/2.0/es/mod/prefork.html#minspareservers), [MaxSpareServers](http://httpd.apache.org/docs/2.0/es/mod/prefork.html#maxspareservers), y [MaxClients](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxclients) regulan la forma en que el proceso padre crea hijos para servir peticiones. En general, Apache funciona bien sin hacer muchas modificaciones en los valores por defecto de estas directivas, de manera que la mayor parte de los sitios web no necesitan ajustar esas directivas a valores diferentes. Los sitios web que necesiten servir más de 256 peticiones simultáneas pueden necesitar incrementar el valor de [MaxClients](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxclients), mientras que los sitios web con memoria limitada pueden necesitar decrementar [MaxClients](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxclients) para evitar que el rendimiento del servidor se degrade (pasando los contenidos de memoria al disco y de vuelta a memoria).

El proceso padre de Apache se inicia normalmente como usuario root en Unix para que escuche en el puerto 80, sin embargo, los procesos hijo se crean con menores privilegios de usuario. Las directivas [User](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#user) y [Group](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#group) se usan para determinar los privilegios de los procesos hijo de Apache. Los procesos hijo deben ser capaces de leer todos los contenidos que van a servir, pero deben tener los menores privilegios posibles.

La directiva [MaxRequestsPerChild](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxrequestsperchild) controla cómo el servidor recicla frecuentemente los procesos eliminando los antiguos y creando nuevos.

**DIRECTIVAS**

* **StartServers:** Número de procesos hijo del servidor que se crean al iniciar Apache. La directiva StartServers especifica el número de procesos hijo que se crean al iniciar Apache. Como el número de procesos está controlado dinámicamente según la carga del servidor, no hay normalmente ninguna razón para modificar el valor de este parámetro.

El valor por defecto cambia según el MPM de que se trate. Para [leader](http://httpd.apache.org/docs/2.0/es/mod/leader.html), [threadpool](http://httpd.apache.org/docs/2.0/es/mod/threadpool.html) y [worker](http://httpd.apache.org/docs/2.0/es/mod/worker.html) el valor por defecto es StartServers 3. Para [prefork](http://httpd.apache.org/docs/2.0/es/mod/prefork.html) el valor por defecto es 5 y para [mpmt\_os2](http://httpd.apache.org/docs/2.0/es/mod/mpmt_os2.html) es 2.

* **MinSpareServers**: Número mínimo de procesos hijo en espera. La directiva MinSpareServers fija el número mínimo de procesos hijo en espera. Un proceso en espera es aquel que no está atendiendo ninguna petición. Si hay menos procesos hijo en espera que MinSpareServers, entonces el proceso padre crea nuevos procesos hijo a un ritmo máximo de uno por segundo.

Ajustar este parámetro debe ser necesario solo en sitios web con muchas visitas. Fijar un valor alto para este parámetro es una mala idea casi siempre.

* **MaxSpareServers**: Número máximo de procesos hijo en espera que puede tener el servidor. La directiva MaxSpareServers determina el número máximo de procesos hijo en espera deseado. Un proceso en espera es aquel que no está atendiendo ninguna petición. Si hay más de MaxSpareServers procesos hijo en espera, entonces el proceso padre elimina el exceso.

Ajustar este parámetro debe ser necesario solo en sitios web con muchas visitas. Fijar un valor alto para este parámetro es una mala idea casi siempre. Si fija un valor por debajo de [MinSpareServers](http://httpd.apache.org/docs/2.0/es/mod/prefork.html#minspareservers), Apache ajustará automáticamente el valor a MinSpareServers + 1.

* **MaxClients**: Número máximo de procesos que serán creados para atender peticiones. La directiva MaxClients especifica el límite de peticiones simultáneas que serán atendidas. Cualquier intento de conexión por encima del límite MaxClients se pondrá en cola, hasta llegar a un límite basado en el valor de la directiva [ListenBacklog](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#listenbacklog). Una vez que un proceso hijo termina de atender una petición y queda libre, se atenderá una conexión en cola.

En servidores que no usan hebras (por ejemplo, [prefork](http://httpd.apache.org/docs/2.0/es/mod/prefork.html)), el valor especificado en MaxClients se traduce en el número máximo de procesos hijo que se crearán para atender peticiones. El valor por defecto es 256; para incrementarlo, debe incrementar también el valor especificado en la directiva [ServerLimit](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#serverlimit).

En servidores que usan hebras y en servidores híbridos (por ejemplo, [beos](http://httpd.apache.org/docs/2.0/es/mod/beos.html) o [worker](http://httpd.apache.org/docs/2.0/es/mod/worker.html)) MaxClients limita el número total de hebras que van a estar disponibles para servir clientes. El valor por defecto para [beos](http://httpd.apache.org/docs/2.0/es/mod/beos.html) es 50. Para MPMs híbridos el valor por defecto es 16 ([ServerLimit](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html" \l "serverlimit)) multiplicado por 25 ([ThreadsPerChild](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html" \l "threadsperchild)). Por lo tanto, si va a usar en MaxClients un valor que requiera más de 16 procesos deberá también incrementar el valor de la directiva [ServerLimit](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#serverlimit).

* **Server Limit**: Límite superior del número configurable de procesos. En el módulo MPM [prefork](http://httpd.apache.org/docs/2.0/es/mod/prefork.html), esta directiva significa el valor máximo que se puede especificar en la directiva [MaxClients](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxclients) sobre el tiempo de vida de un proceso de Apache. Los intententos de cambiar el valor de esta directiva durante el reinicio del servidor serán ignorados. El valor de [MaxClients](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxclients) sí que puede modificarse durante el reinicio.

Cuando se usa esta directiva hay que tener especial cuidado. Si en la directiva ServerLimit se especifica un valor mucho más alto de lo necesario, se reservará memoria compartida que no será usada. Si ambas directivas ServerLimit y [MaxClients](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxclients) tienen especificados valores mayores que los que el sistema puede manejar, Apache puede que no se inicie o que el sistema se vuelva inestable.

Con el módulo MPM [prefork](http://httpd.apache.org/docs/2.0/es/mod/prefork.html), use esta directiva solamente si necesita especificar en la directiva [MaxClients](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxclients) un valor mayor a 256 (el valor por defecto). No especifique un valor mayor del que vaya a especificar en la directiva [MaxClients](http://httpd.apache.org/docs/2.0/es/mod/mpm_common.html#maxclients).

### Nota

Existe un límite inviolable compilado en el servidor que es ServerLimit 20000. Con este límite se intentan evitar las consecuencias que pueden tener los errores tipográficos.

* **MaxRequestsPerChild**: Límite en el número de peticiones que un proceso hijo puede atender durante su vida.

La directiva MaxRequestsPerChild especifica el número máximo de peticiones que un proceso hijo atenderá durante su existencia. Después de atender MaxRequestsPerChild peticiones, el proceso hijo se eliminará. Si el valor especificado en esta directiva MaxRequestsPerChild es 0, no habrá límite.

Especificar en la directiva MaxRequestsPerChild un valor distinto de cero tiene dos ventajas:

* limita la cantidad de memoria que un proceso puede consumir en caso de que haya un fuga (accidental) de memoria;
* establece un límite finito a la vida de los procesos, lo que ayuda a reducir el número existente de procesos cuando se reduce la carga de trabajo en el servidor.

*Para probar el rendimiento del apache, puedes hacer un test que lleva incluido el propio apache. Es un ejecutable que se llama "ab" y está en el directorio /bin o /sbin. Este programa simula gran cantidad de tráfico para el Apache y es una muy buena manera de medir el rendimiento de tu apache.*

# [Consultar Directrices de configuración en httpd.conf](http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/s1-apache-config.html)

# [Optimización de apache](http://www.profesionalhosting.com/soporte-en-linea/optimizacin-de-apache-preg67.html)