

PHP-FPM Optimization

Out-of-box php fpm is configured for very low server specs such as a 2 core machine. It needs to be configured to match the hardware you are on. You need to factor on the most expensive processes you run.

Typically a low-end production server has 4 cores with 8 GB RAM so you can use the following configuration:

Edit the file `/etc/apache2/mods-enabled/mpm-event.conf` and add the following:

```
# event MPM

# StartServers: initial number of server processes to start
# MinSpareThreads: minimum number of worker threads which are kept spare
# MaxSpareThreads: maximum number of worker threads which are kept spare
# ThreadsPerChild: constant number of worker threads in each server process
# MaxRequestWorkers: maximum number of worker threads
# MaxConnectionsPerChild: maximum number of requests a server process serves
# <IfModule mpm_event_module>
#   StartServers 2
#   MinSpareThreads 25
#   MaxSpareThreads 75
#   ThreadLimit 64
#   ThreadsPerChild 25
#   MaxRequestWorkers 150
#   MaxConnectionsPerChild 0
# </IfModule>

# ServerLimit      (Total RAM - Memory used for Linux, DB, etc.) / process size
# StartServers     (Number of Cores)
# MaxRequestWorkers (Total RAM - Memory used for Linux, DB, etc.) / process size

<IfModule mpm_event_module>
    # for c5 classes with only 8GB ram
    # ServerLimit      500
    StartServers      4
    MinSpareThreads    25
    MaxSpareThreads    75
    ThreadLimit        64
```

```
ThreadsPerChild      25
MaxRequestWorkers     2800
# for c5 classes with only 8GB ram
# MaxRequestWorkers   1400
MaxConnectionsPerChild 1000
</IfModule>
```

Edit the file `/etc/php/7.4/fpm/pool.d/www.conf` and make sure the following settings are there:

```
; settings explanation - don't need to copy this
;pm.max_children      (total RAM - (DB etc) / process size)
;pm.start_servers      (cpu cores * 4)
;pm.min_spare_servers  (cpu cores * 2)
;pm.max_spare_servers  (cpu cores * 4)

; default is dynamic but that can churn up the memory because it leaves processes lingering
; pm = dynamic
pm = ondemand
; default is pm.max_children = 5
pm.max_children = 256

; everything below is only relevant if using pm = dynamic
; for c class servers with only 8GB ram
; pm.max_children = 128
; default is pm.start_servers = 2
pm.start_servers = 16
; default is pm.min_spare_servers = 1
pm.min_spare_servers = 8
; default is pm.max_spare_servers = 3
pm.max_spare_servers = 16
; setting to 0 or leaving commented out will use the PHP_FCGI_MAX_REQUESTS value whatever that is.
pm.max_requests = 1000
```

Now we have allowed php to run a lot more threads we may run into a “too many open files” error.

To fix edit `/etc/php/7.4/fpm/php-fpm.conf` and change the `rlimit_files` to 4096. If you are still getting the “too many open files” error you can double this.

```
rlimit_files = 10000
```

You can also try editing `/etc/security/limits.conf` and adding the following:

```
*          hard  nofile   10000
*          soft  nofile   10000
www-data   soft  nofile   10000
www-data   hard  nofile   10000
```

Restart everything:

```
sudo service apache2 restart && sudo service php7.4-fpm restart
```

See also <https://medium.com/@sbuckpesch/apache2-and-php-fpm-performance-optimization-step-by-step-guide-1bfecf161534>

Revision #1

Created 14 June 2024 00:59:28 by peterd

Updated 14 June 2024 00:59:43 by peterd