#############

Php-fpm.conf

#############

;;;;;;;;;;;;;;;;;;;;

; FPM Configuration ;

;;;;;;;;;;;;;;;;;;;;;

; All relative paths in this configuration file are relative to PHP's install

; prefix (/usr). This prefix can be dynamically changed by using the

; '-p' argument from the command line.

;;;;;;;;;;;;;;;;;;

; Global Options ;

;;;;;;;;;;;;;;;;;;

[global]

; Pid file

; Note: the default prefix is /var

; Default Value: none

pid = /run/php/php81-fpm.pid

; Error log file

; If it's set to "syslog", log is sent to syslogd instead of being written

; in a local file.

; Note: the default prefix is /var

; Default Value: log/php-fpm.log

error\_log = /proc/self/fd/2

; syslog\_facility is used to specify what type of program is logging the

; message. This lets syslogd specify that messages from different facilities

; will be handled differently.

; See syslog(3) for possible values (ex daemon equiv LOG\_DAEMON)

; Default Value: daemon

;syslog.facility = daemon

; syslog\_ident is prepended to every message. If you have multiple FPM

; instances running on the same server, you can change the default value

; which must suit common needs.

; Default Value: php-fpm

;syslog.ident = php-fpm

; Log level

; Possible Values: alert, error, warning, notice, debug

; Default Value: notice

;log\_level = notice

; If this number of child processes exit with SIGSEGV or SIGBUS within the time

; interval set by emergency\_restart\_interval then FPM will restart. A value

; of '0' means 'Off'.

; Default Value: 0

;emergency\_restart\_threshold = 0

; Interval of time used by emergency\_restart\_interval to determine when

; a graceful restart will be initiated. This can be useful to work around

; accidental corruptions in an accelerator's shared memory.

; Available Units: s(econds), m(inutes), h(ours), or d(ays)

; Default Unit: seconds

; Default Value: 0

;emergency\_restart\_interval = 0

; Time limit for child processes to wait for a reaction on signals from master.

; Available units: s(econds), m(inutes), h(ours), or d(ays)

; Default Unit: seconds

; Default Value: 0

;process\_control\_timeout = 0

; The maximum number of processes FPM will fork. This has been design to control

; the global number of processes when using dynamic PM within a lot of pools.

; Use it with caution.

; Note: A value of 0 indicates no limit

; Default Value: 0

; process.max = 128

; Specify the nice(2) priority to apply to the master process (only if set)

; The value can vary from -19 (highest priority) to 20 (lower priority)

; Note: - It will only work if the FPM master process is launched as root

; - The pool process will inherit the master process priority

; unless it specified otherwise

; Default Value: no set

; process.priority = -19

; Send FPM to background. Set to 'no' to keep FPM in foreground for debugging.

; Default Value: yes

daemonize = no

; Set open file descriptor rlimit for the master process.

; Default Value: system defined value

;rlimit\_files = 1024

; Set max core size rlimit for the master process.

; Possible Values: 'unlimited' or an integer greater or equal to 0

; Default Value: system defined value

;rlimit\_core = 0

; Specify the event mechanism FPM will use. The following is available:

; - select (any POSIX os)

; - poll (any POSIX os)

; - epoll (linux >= 2.5.44)

; - kqueue (FreeBSD >= 4.1, OpenBSD >= 2.9, NetBSD >= 2.0)

; - /dev/poll (Solaris >= 7)

; - port (Solaris >= 10)

; Default Value: not set (auto detection)

;events.mechanism = epoll

; When FPM is build with systemd integration, specify the interval,

; in second, between health report notification to systemd.

; Set to 0 to disable.

; Available Units: s(econds), m(inutes), h(ours)

; Default Unit: seconds

; Default value: 10

;systemd\_interval = 10

;;;;;;;;;;;;;;;;;;;;

; Pool Definitions ;

;;;;;;;;;;;;;;;;;;;;

; Multiple pools of child processes may be started with different listening

; ports and different management options. The name of the pool will be

; used in logs and stats. There is no limitation on the number of pools which

; FPM can handle. Your system will tell you anyway :)

; Include one or more files. If glob(3) exists, it is used to include a bunch of

; files from a glob(3) pattern. This directive can be used everywhere in the

; file.

; Relative path can also be used. They will be prefixed by:

; - the global prefix if it's been set (-p argument)

; - /usr otherwise

include=/etc/php81/php-fpm.d/\*.conf