Apache Server Day 2

Mohamed Emary

January 21, 2025

1 Worker Vs Event Vs Prefork MPM

1.1 Worker MPM

The Worker Multi-Processing Module (MPM) implements a hybrid multi-process, multi-threaded server. By using threads to handle requests, it can serve a large number of requests with fewer system resources compared to a purely process-based server.

This MPM maintains much of the stability of a process-based server by keeping multiple processes available, each containing many threads.

Advantages of Worker MPM:

- Being able to serve more requests
- Lower resources consumption

Disadvantages of Worker MPM:

- If a thread crashes, the whole process crashes

1.2 Event MPM

This Multi-Processing Module (MPM) implements a hybrid multi-process multi-threaded server.

By using threads to serve requests, it is able to serve a large number of requests with fewer system resources than a process-based server.

It retains much of the stability of a process-based server by keeping multiple processes available, each with many threads.

1.3 Prefork MPM

The Prefork Multi-Processing Module (MPM) implements a non-threaded, pre-forking web server. Each server process handles incoming requests, while a parent process manages the server pool size. This MPM is suitable for sites that need to avoid threading due to compatibility issues with non-thread-safe libraries.

Additionally, Prefork MPM is ideal for isolating each request, ensuring that a problem with one request does not impact others.

```
<IfModule mpm_prefork_module>
  StartServers 5
```

```
  MinSpareServers 5
  MaxSpareServers 10
  MaxClients 150
</IfModule>
```

1.4 Apache MPM Comparison: Prefork vs. Worker vs. Event

Feature	Prefork MPM	Worker MPM	Event MPM
Process Management	Multiple child processes	Multiple child processes with threads	Multiple child processes with threads
Threads	Single thread per process	Multiple threads per process	Multiple threads per process
Memory Usage	High (each process uses more memory)	Lower (threads share memory)	Lower (threads share memory)
Performance	Good for low-traffic sites	Better for high-traffic sites	Best for high-traffic sites
Compatibility	High compatibility with PHP	Requires thread-safe modules	Requires thread-safe modules
Stability	More stable (less complex)	Slightly less stable (more complex)	Similar stability to Worker MPM
Scalability	Limited scalability	Good scalability	Excellent scalability
Keep-Alive Handling	Basic keep-alive support	Better keep-alive support	Advanced keep-alive support

1.4.1 Advantages and Disadvantages

MPM	Advantages	Disadvantages
Prefork	High compatibility with PHP, simple configuration	High memory usage, limited scalability
Worker	Efficient memory usage, good scalability	Requires thread-safe modules, slightly less stable
Event	Excellent scalability, advanced keep-alive support, efficient memory usage	Requires thread-safe modules, more complex configuration