Dashboard de performances – Serveur TCP/HTTP

Types de serveurs présents : mono, multi. Plage de charge testée : de 10 à 300 clients simultanés.

Analyse avancée

Globalement, les mesures de benchmark montrent que le serveur multi-thread offre un débit moyen d'environ 129.3 requêtes par seconde, contre 14.8 req/s pour le serveur mono-thread. Sur l'ensemble des configurations testées, cela correspond à un gain moyen de performance d'environ 8.76× en faveur de l'architecture multi-thread.

En termes de latence, la mesure P99 (latence subie par les 1 % de requêtes les plus lentes) reste plus favorable au serveur multi-thread, avec une P99 moyenne de 858.9 ms contre 4011.5 ms pour le mono-thread. Cela indique que le multi-thread absorbe mieux les pics de charge et réduit les phénomènes de saturation lorsque le nombre de clients simultanés augmente.

À la charge la plus élevée (≈ 300 clients), on observe un débit de 140.9 req/s pour le serveur multi-thread contre 9.2 req/s pour le mono-thread. La latence P99 atteint 5938.0 ms côté mono, alors qu'elle est de 1925.2 ms côté multi, ce qui confirme que le mono-thread atteint rapidement un plateau de performance tandis que le multi-thread continue à exploiter les cœurs CPU disponibles.

L'analyse de l'utilisation CPU montre que les deux architectures finissent par saturer les cœurs disponibles, mais le serveur multi-thread parvient à transformer cette consommation CPU en débit utile plus élevé (CPU moyen ≈ 1.0 % contre 0.1 % pour le mono-thread). La consommation mémoire reste globalement maîtrisée pour les deux serveurs, avec une légère surconsommation attendue côté multi-thread liée à la gestion des threads et de la file FIFO.

En pratique, l'architecture multi-thread avec file FIFO bornée constitue le meilleur choix pour un environnement de production soumis à des pics de charge importants, à condition de maîtriser la complexité de synchronisation et l'arrêt propre des threads. Le serveur mono-thread conserve néanmoins un intérêt pédagogique fort et peut être adapté à des scénarios simples ou à faible charge, où la lisibilité du code prime sur la performance brute.

1-throughput

2-latency_p99

3-cpu

4-memory

5-speedup