

ABSTRACT

Augmenting the web server's capacity may lead to system failures or excessive load, ultimately resulting in server downtime. In order to address this issue, a load-balancing strategy is employed, which enhances the scalability and efficiency of servers. This technique involves the distribution of incoming requests over multiple servers simultaneously, thereby reducing the workload absorbed by each individual server. The scalability of the system enables the equitable distribution of server traffic loads across numerous connection lines, optimizing throughput, enhancing reaction time, and mitigating the risk of overload on any single connection line. This study aims to conduct a load-balancing experiment using the Raspberry Pi 3B+ device. The Nginx load balancer program will be utilized, employing the Round Robin and Least Connection algorithms. These methods will be connected to two Django web servers. The metrics to be evaluated for comparison include throughput, delay, reaction time, CPU use, and RAM utilization. In the analysis of the Throughput, Delay, and Response time metrics for the two algorithms, it was observed that there were no statistically significant differences, with the exception of the tests conducted with 5000 and 6000 requests. These particular tests yielded inconclusive findings due to mistakes encountered during the execution of the two algorithms. Therefore, it can be inferred that the utilization of Round-Robin and Least Connection algorithms is appropriate for a website experiencing traffic volumes ranging from 500 to 4000 requests per minute, with content sizes that are not excessively large, in order to prevent any potential issues during the server load sharing procedure.

Keywords : Raspberry Pi 3B+, Load Balancing, Nginx, Django, Round robin, Least Connection