Increasing web server performance using the web balancing method

I Saraswati*, S Praptodiyono, A S Pramudyo and Kurniawan
Electrical Engineering Department, Universitas Sultan Ageng Tirtayasa, Jl. Jend. Sudirman Km. 3, Cilegon, Indonesia

*Email: irma.saraswati@untirta.ac.id

Abstract. The high level of activity of internet users, which will have a negative impact on information providers. Requests from many users can cause performance capabilities on the web-server to decrease. The large number of requests results in crashes or overloads that have an impact on requests that cannot be served by a single server. Web balancing technology is needed so that the incoming request load is not only served by one server. Some scheduling algorithms that will be applied to web balancing testing in this study are round-robin and weight round-robin. In this study using ipvsadm as a means of making web balancing. This test aims to determine the performance of web servers when using web balancing technology by measuring the QoS parameters of web server services. The results show, the best value for measuring QoS parameters is response time, request error, throughput, in the weight round-robin scheduling algorithm method.

1. Introduction
Today, the internet has become a very important requirement for everyone, especially for people who live in urban areas, who have been very literate with technological developments. Every day people will always access the internet in any form. According to the latest report from the Indonesian Internet Service Providers Association (APJII) where 64.8% or around 171.17 million people from 264 million people at 2019 in Indonesia have been connected to the internet network [1, 2].

Seeing the high level of activity of internet users, which will have a negative impact on information providers. So that it can cause performance capabilities on a web-server can be reduced because of many requests from internet users. The large number of requests from users can crash or overload the possibility that the web server will be down due to too many requests [3].

Web server is a software that functions to receive requests for Hypertext Transfer Protocol (HTTP) from users who access the internet with a web browser and provide or send back the results of user requests that are usually in the form of web pages, which generally display information such as text, video, image, animation and audio data [4]. The web server works because of a request or HTTP request from the client and replies to it by providing content in the form of a web page. Most of the number of requests that can be handled by a web server depends on the specifications of the web server. With increasing numbers of internet users who often access, a web server will experience overload, then only one server is not enough to handle too many requests from the client [5].

To overcome the problem that occurs because the advantage is to upgrade the specifications on the web-server, but increasing specifications on web-servers requires a very large budget and usually, only large companies can spend high budgets to get high specifications and good web server performance,
but what about small companies or organizations that do not have a large budget to upgrade specifications on web-servers, this problem is the most important for the implementation of load balancing technology [6, 7].

Load balancing technology is a technique for distributing traffic loads on two or more connection paths in a balanced manner, so that traffic runs optimally and avoids overloading of connection lines. Load balancing is a system that can service large access loads with the possibility of reducing failure when serving client requests. Load balancing technology is basically to know how to use the web server effectively, so that computer networks can work well without experiencing obstacles [8]. Load balancing works to accept requests from clients, when the request is received then load balancing is usually cleverly able to determine which server has lower request load and faster response times.

Research conducted by Faris P.P [9], about load balancing analysis with the round robin method, where this method works to divide the traffic that goes evenly to each web server, while the research conducted by Dany R [10], about web server load balancing analysis using the least connection method, which divides the load by giving requests to servers that have less traffic activity. In the previous study [9] [10] carried out in a simulation by using Software Define Network and only performed performance tests on one of the scheduling algorithms available in the LVS technique. So that in this study a simple network load balancing web server will be created to compare the performance of web servers, when using several scheduling algorithms that exist in load balancing techniques and when a single server.

2. Load Balancing

Load Balancing is a process or technology that distributes the traffic of a site to several servers using a network device, the device receives traffic aimed at a site and distributes it to several servers. Load balancing can be implemented with special hardware, software or a combination of both. The existing standard configuration illustrates that one machine is placed between the client and server, this machine is called the director because the task is to provide balancing on requests from the client to the server.

This study was explores information from previous research studies as reference material, both regarding the advantages and disadvantages that exist.

a. Research title "Analysis of Load Balancing in Web Servers Using the Least Connection Algorithm" [10] discussed load balancing with LVS direct routing topology and uses the least connection scheduling algorithm. In this study the web server is made virtually by using the piranha tool that is on the Linux operating system.

b. The research title “Load balancing Performance Analysis with Round Robin Algorithm in Software Defined Network (SDN)” [9]. This research deals with load balancing technology using a round robin scheduling method that is applied and simulated on software defined networks

3. Research Methods

Broadly speaking, research is carried out in stages as will be explained as follows:

a. Conduct literature studies so that can determine the steps for designing the system.

b. General plant design divided into 2, namely hardware design (hardware) and software design (software). The next stage determines what components are needed both in software and software in web balancing.

c. Realize the system in accordance with the design and analysis that has been done.

d. The next stage is testing web balancing technology.

e. Testing is done by giving a different burden on each experiment after the implementation of web balancing and before the implementation of web balancing.

f. Overall system testing is done to determine server performance when using web balancing technology with two scheduling methods and methods without web balancing.

g. The final step is the analysis of system errors and their causes.

h. Conclusions.

3.1. Research flow chart
In simple terms, the process of designing a local network enhancing server performance with the implementation of a web balancing system is illustrated through a flow diagram in figure 1.

![Research Flow Chart](image)

**Figure 1.** Research Flow Chart.

3.2. Research design
The design carried out involves all research instruments, which are connected to a web server, director or load balancer that plays a role in sharing the load on the server, the load balancing process occurs at the Director connected to the switch for later continue the request from the user to the web server. The design is done with two web servers with one web balance, where the web server has been subsonic installed as content on the web server and the web balancer has been installed ipvsadm which functions for the balancing web process. Where Ethernet 1 on the web balancer device has been connected to two subsonic and Ethernet 2 installed servers connected to the user who will access the web server. The overall installation can be seen in figure 2 below.

![Topology web balancing](image)

**Figure 2.** Topology web balancing.

4. Results and Discussion
4.1. Results of application of web balancing
The result of the application of web balancing technology is that it has implemented web balancing technology that is able to evenly distribute the load according to server specifications when the server receives requests from users. Where the application of web balancing technology in this research applies two scheduling algorithms namely round-robin and weight round-robin. The results of the web balancing technology that has been implemented that are able to evenly distribute the load to each server can be seen in figure 3.

![Figure 3. Load Balancing metode penjadwalan round-robin.](image)

In figure 3 is web balancing technology that has been applied with a round-robin scheduling algorithm wherein this algorithm distributes requests that enter evenly to each server.

4.2. Results of application of web balancing
The results of testing the web balancing technology is done by giving a load to the server with different values, namely 1000 request/s, 1500 request/s, 2000 request/s and 5000 requests/s. do testing before the web balancing method is implemented with two round-robin scheduling methods and round-robin weight and compare it to the situation without web balancing technology. The compared parameters include response time, request error, and throughput, the experiment was carried out ten times in each method that was applied. The results of testing on each method that has been applied can be seen in table 1 for the average value of response time, table 2 for the request error average value and table 3 for the average throughput value.

4.2.1. Response time testing results
Response time is the time interval between the user entering a command into the system to get a response or response from a system that has received a request from the client and how quickly the system responds to commands from the user. The faster the response time value, the faster the web display is visited by the user. Table 1 is the result of the average response time of each test that has been carried
out, after the implementation of web balancing technology with two round-robin scheduling and weight round-robin scheduling and testing before the implementation of web balancing technology. Tests are carried out with 4 scenarios, namely by giving a request load of 1000, 1500, 2000 and 5000 requests/s.

### Table 1. Average Results of Response Time Testing.

| Load   | Single Server (ms) | Round-robin (ms) | Weight Round-robin (ms) |
|--------|--------------------|------------------|-------------------------|
| 1000   | 3.66               | 1.56             | 1.16                    |
| 1500   | 9.99               | 3.82             | 2.64                    |
| 2000   | 31.55              | 5.37             | 4.92                    |
| 5000   | 82.44              | 47.79            | 28.75                   |

In each request load scenario, the round-robin weight scheduling algorithm receives the fastest response time value compared to the round-robin scheduling algorithm and single server with a value of 1.16 ms, 2.64 ms, 4.92 ms, and 28.75 ms. This is due to the weight round-robin scheduling algorithm dividing the load according to the capabilities and specifications of each server, where servers with high specifications will receive a greater load than servers that have lower specifications, so the server works more optimally.

#### 4.2.2. Test results for error request

The request error is as much as the number of requests from a user that is not able to be served by the server, which means that the smaller the value of the request error, the more reliable the performance of the web server is to serve any request from the user. To see the experimental average value of each method can be seen in table 2. Table 2 is the result of the average error request of each test that has been carried out, after the implementation of web balancing technology with two round-robin scheduling and weight round-robin scheduling and testing before the implementation of web balancing technology. Tests are carried out with 4 scenarios, namely by giving a request load of 1000, 1500, 2000 and 5000 requests/s.

### Table 2. Average Results for Testing Error Request.

| Load   | Single Server (%) | Round-robin (%) | Weight Round-robin (%) |
|--------|-------------------|-----------------|------------------------|
| 1000   | 0.12              | 0.03            | 0.02                   |
| 1500   | 0.38              | 0.07            | 0.05                   |
| 2000   | 2.48              | 0.99            | 0.30                   |
| 5000   | 3.06              | 1.10            | 1.03                   |

In each scenario of requesting burdens that have been carried out web balancing technology with a weight round-robin scheduling algorithm, obtain the lowest error request value compared to the round-robin scheduling algorithm and a single server with a value of 0.02%, 0.05%, 0, 30% and 1.03%, this is due to the weight round-robin scheduling algorithm dividing the load according to the capabilities and specifications of each server.

#### 4.2.3. Net I/O testing results (throughput)

The Net I/O value in this study represents the number of users requests that the web server can respond to in one-time unit. The greater the Net I/O value or throughput generated shows that the web server's performance is getting better because more and more user requests can be responded by a web server in a one-time unit. To see the average value of Net I/O (Throughput) can be seen in table 3.

### Table 3. Average Results of Net I/O Testing (Throughput).
| Load  | Single Server (KB/s) | Round-robin (KB/s) | Weight Round-robin (KB/s) |
|-------|----------------------|-------------------|---------------------------|
| 1000  | 276.14               | 390.03            | 416.99                    |
| 1500  | 440.11               | 700.29            | 469.38                    |
| 2000  | 494.33               | 738.71            | 833.3                     |
| 5000  | 694.84               | 906.46            | 923.95                    |

Table 3 is the result of the average throughput of each test that has been carried out, after the implementation of web balancing technology with two round-robin scheduling and weight round-robin and testing algorithms before the implementation of web balancing technology. Tests are carried out with 4 scenarios, namely by giving a request load of 1000, 1500, 2000 and 5000 requests/s.

On the 4 scenarios for providing the burden of requests that have been carried out the highest Net I/O (throughput) value is obtained from the scheme after the implementation of web balancing technology by applying the weight round-robin scheduling algorithm with a value of 416.99 KB/s, 469.38 KB/s, 833.3 KB/s and 923.95 KB/s. This is due to the weight round-robin scheduling method that responds to all requests from the user which can be seen in Table 2.

5. Conclusion
Based on web balancing testing on two methods, namely rr and wrr, it was found that the performance or performance of the server increased compared to the single server scheme and the incoming traffic load was distributed equally to each server. In the response time parameter, web balancing with the weight round-robin scheduling method gets the fastest value of 1.16 ms, 2.64 ms, 4.92 ms, and 28.75 ms compared to the other two methods. Testing on error request parameters, web balancing with the round-robin weight scheduling method provides a fairly small error value of 0.02%, 0.05%, 0.30%, and 1.03%. Compared to the other two methods. Furthermore, through the Throughput parameter, web balancing with the round-robin weight scheduling method received the highest values of 416.99, 469.4, 833.3 and 923.95 compared to the other two methods. Based on the description of the tests that have been done, it is found that the wrr scheduling method is more suitable to be applied when the hardware specifications used by the web server are different.

6. Reference
[1] Alsyaib O M A and Sukardiyono T 2014 Jurnal Elektronik Pendidikan Teknik Informatika 3 1
[2] Schaar V D 2019 Research Report: A Deeply Divided Nation (Jakarta: Indonesia Investments)
[3] Meena K, Amuthadevi C, and Kumar G S 2018 International Journal of Engineering & Technology 7 (2.4) 85-87
[4] Aaqib S M and Sharma L 2013 International Journal of Applied Information Systems 5 3 42-46
[5] Menasce D A 2003 IEEE Internet Computing 7 6
[6] Julianto R, Yahya W, and Akbar S R 2017 J. Pengembangan Teknologi Informasi dan Ilmu Komputer 1 9 904-914
[7] Shrivastava G, Kaushik P, and Pateriya P 2018 Int. J. of Eng.and Tech. 7 (3) 1854-1857
[8] Pandey S, Prasanna S, Kapil S, and Rajeshwari B S 2015 Int. J. of Adv Research in Comp. Sci and Management Studies 3 331-335
[9] Perdana F P, Irawan B, and Latuconsina R 2017 eProceeding Eng. of Telkom Univ. 4 3
[10] Dany R, Primananda R, and Yahya W 2018 J. Pengembangan Teknologi Informasi dan Ilmu Komputer 2 3 915-920