



**ANALISIS *QUALITY OF SERVICES* ALGORITMA IP  
*HASH, ROUND ROBIN* DAN *LEAST CONNECTION*  
SEBAGAI *LOAD BALANCE* PADA LAYANAN *WEB*  
*SERVER* BERBASIS *MICROSERVICES***

**TESIS**

**OLEH**  
**ROY RINALDI**  
**55419110010**

**PROGRAM MAGISTER TEKNIK ELEKTRO**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS MERCU BUANA**  
**2021**



**ANALISIS *QUALITY OF SERVICES* ALGORITMA IP  
*HASH, ROUND ROBIN* DAN *LEAST CONNECTION*  
SEBAGAI *LOAD BALANCE* PADA LAYANAN *WEB*  
*SERVER* BERBASIS *MICROSERVICES***

**TESIS**

**Diajukan sebagai Salah Satu Syarat untuk Menyelesaikan**

**Program Studi Magister Teknik**

**UNIVERSITAS**

**MERCU BUANA**

**OLEH**

**ROY RINALDI**

**55419110010**

**PROGRAM MAGISTER TEKNIK ELEKTRO**

**FAKULTAS TEKNIK**

**UNIVERSITAS MERCU BUANA**

**2021**

## ABSTRAK

Era serba teknologi saat ini *internet* menjadi salah satu sumber dalam mencari informasi, akan tetapi terkadang suatu *website* tidak dapat diakses dikarenakan *overload* akibat *traffic* yang tinggi. Hal tersebut tentunya membutuhkan mekanisme untuk mengatur beban pada *server*, salah satunya adalah dengan menggunakan *load balance*.

Pada penelitian sebelumnya penggunaan algoritma *round robin*, *least connection*, dan *IP hash* sebagai *load balance* pada *web server* dapat mengurangi beban pada *web server* dengan baik, namun masih memiliki kekurangan dari sisi penggunaan *resource* yang kurang efisien. Dalam penelitian ini diusulkan penggunaan algoritma tersebut dengan pengembangan *web server* secara *microservices* agar performa *web server* lebih dapat optimal dan efisien dalam penggunaan *resource*.

Dari serangkaian pengujian yang telah dilakukan dengan memberi beban pada *web server* kemudian diukur menggunakan parameter *throughput*, *jitter*, *delay/latency* serta *resource utilization*. Berdasarkan standarisasi *Quality of Services* yang mengacu pada TIPHON algoritma ketiga algoritma memiliki performa yang baik. Hasil skenario pertama untuk pengujian CPU, algoritma *round robin* memiliki performa yang lebih baik dibandingkan lainnya yaitu 22,3%. Sedangkan untuk efisiensi *memory* algoritma *IP hash* hanya mengkonsumsi *memory* 7,6% dari total *memory*. Kemudian untuk pengujian *throughput*, algoritma *IP hash* memiliki performa diatas algoritma lainnya dengan rata-rata *throughput* 443 KB/s. Pengujian *delay* dan *jitter* algoritma *IP hash* memiliki performa yang lebih baik dimana *delay* yang dihasilkan adalah 0,19 ms dan *jitter* 0,39 ms. Pada skenario kedua dengan pengujian 30.000 *http request*, algoritma *IP hash* memiliki performa yang lebih baik dengan rata – rata CPU 12,3 %, *memory* 6,45 %, *throughput* 225 KB/s, *delay* 0,22 ms dan *jitter* 0,44 ms. Lalu pada pengujian ketiga dengan *IP hash switch over* memerlukan waktu selama 10 *second* untuk mengarahkan *request* ke node-2.

*Keyword* : *Overload, Load Balance, Web Server, Microservices, Quality of Services*

## ABSTRACT

*In this era of all-technology, the internet is one of the sources for finding information, but sometimes a website cannot be accessed due to overload due to high traffic. This situation requires a mechanism to regulate the load on the server, one of which is by using a load balance. In previous studies, the use of the round robin algorithm, least connection, and IP hash as a load balance on the web server can reduce the load on the web server well, but still has shortcomings in terms of inefficient use of resources. In this study, it is proposed to use the algorithm with microservices web server development so that web server performance can be optimized and efficient in resource use. From a series of tests that have been carried out by loading the web server, then it is measured using the parameters of throughput, jitter, delay/latency and resource utilization. Based on the standardization of Quality of Services which refers to the TIPHON algorithm, the three algorithms have good performance. The results of the first scenario for CPU testing, the round robin algorithm has a better performance than the others, which is 22.3%. Meanwhile, for memory efficiency, the IP hash algorithm only consumes 7.6% of the total memory. Then for throughput testing, the IP hash algorithm has performance above other algorithms with an average throughput of 443 KB/s. The delay and jitter test of the IP hash algorithm has better performance where the resulting delay is 0.19 ms and the jitter is 0.39 ms. In the second scenario by testing 30,000 http requests, the IP hash algorithm has better performance with an average CPU of 12.3%, memory 6.45%, throughput 225 KB/s, delay 0.22 ms and jitter 0.44 ms. Then in the third test with an IP hash switch over it takes 10 seconds to direct the request to node-2.*

*Keyword : Overload, Load Balance, Web Server, Microservices, Quality of Services*

## PENGESAHAN TESIS

Judul : Analisis *Quality Of Services* Algoritma IP Hash, Round Robin Dan Least Connection Sebagai Load Balance Pada Layanan Web Server Berbasis *Microservices*

Nama : Roy Rinaldi

NIM : 55419110010

Program Studi : Magister Teknik Elektro

Tanggal : 14 Agustus 2021

Mengesahkan

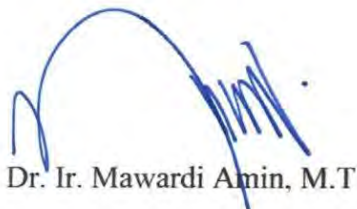
Pembimbing



Dr. Marza Ihsan Marzuki M.T.

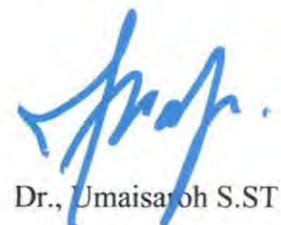
UNIVERSITAS  
MERCU BUANA

Dekan Fakultas Teknik



Dr. Ir. Mawardi Amin, M.T.

Ketua Program Studi  
Magister Teknik Elektro



Dr., Umairah S.ST

## PERNYATAAN *SIMILARITY CHECK*

Saya yang bertanda tangan di bawah ini menyatakan, bahwa karya ilmiah yang ditulis oleh

Nama : Roy Rinaldi  
NIM : 55419110010  
Program Studi : Magister Teknik Elektro

dengan judul

*“An Analysis Quality of Service Load Balance Algorithm on Web server With Microservices Architecture”*,

telah dilakukan pengecekan *similarity* dengan sistem Turnitin pada tanggal 06/08/21, didapatkan nilai persentase sebesar 14 %.

Jakarta, 6 Agustus 2020  
Administrator Turnitin

  
Arie Pangudi, A.Md

UNIVERSITAS  
MERCU BUANA



## PERNYATAAN

Saya yang bertandatangan dibawah ini menyatakan dengan sebenar-benarnya bahwa semua pernyataan dalam tesis ini:

Judul : ANALISIS QUALITY OF SERVICES ALGORITMA IP HASH, ROUND ROBIN DAN LEAST CONNECTION SEBAGAI LOAD BALANCE PADA LAYANAN WEB SERVER BERBASIS MICROSERVICES  
Nama : Roy Rinaldi  
NIM : 55419110010  
Program Studi : Magister Teknik Elektro  
Tanggal : 13 Agustus 2021

Merupakan hasil studi pustaka, penelitian lapangan dan karya saya sendiri dengan bimbingan Komisi Dosen Pembimbing yang ditetapkan dengan Surat Keputusan Ketua Program Studi Magister Teknik Elektro Universitas Mercu Buana.

Karya ilmiah ini belum pernah diajukan untuk memperoleh gelar kesarjanaan pada program sejenis di perguruan tinggi lain. Semua informasi, data dan hasil pengolahannya yang digunakan telah dinyatakan secara jelas sumbernya dan dapat diperiksa kebenarannya.

UNIVERSITAS  
MERCU BUANA

Jakarta, 13 Agustus 2021



Roy Rinaldi

## KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah memberikan Rahmat dan Hidayah-Nya sehingga penyusunan Tesis ini bisa diselesaikan sebagai syarat menyelesaikan pendidikan Magister Teknik Elektro pada Program Pascasarjana Universitas Mercu Buana dengan judul “*Analisis Quality Of Services Algoritma IP Hash, Round Robin Dan Least Connection Sebagai Load Balance Pada Layanan Web Server Berbasis Microservices*”.

Penulis menyadari bahwa tanpa adanya bantuan dan bimbingan dari berbagai pihak yang selama ini turut membantu penulis mulai dari masa perkuliahan hingga pada penyusunan Thesis ini, penulis akan mengalami kesulitan. Oleh karena itu pada kesempatan kali ini kiranya penulis ingin mengucapkan terima kasih banyak kepada semua pihak, diantaranya:

1. Dr. Marza Ihsan Marzuki, M.T. Selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penulis dalam penyusunan laporan Tesis ini sampai dengan selesai.
2. Ibu Dr., Umairah S.ST selaku Ketua Program Studi Magister Teknik Elektro Universitas Mercubuana yang telah memberikan arahan, pedoman, serta motivasi sehingga penulisan Tesis ini dapat diselesaikan dengan baik.
3. Orang Tua dan Kerabat dekat penulis yang telah memberikan bantuan dukungan moril maupun materil hingga penulis dapat menyelesaikan Tesis ini.
4. Rekan – rekan mahasiswa Universitas Mercu Buana yang memberikan arahan serta *sharing knowledge* sebagai bahan penyusun Tesis ini.

Penulis menyadari tesis ini masih jauh dari sempurna, oleh karena itu penulis menerima segala kritik dan saran yang membangun demi untuk perbaikan tesis ini.

Jakarta, 12 Juni 2021

Penulis



## DAFTAR ISI

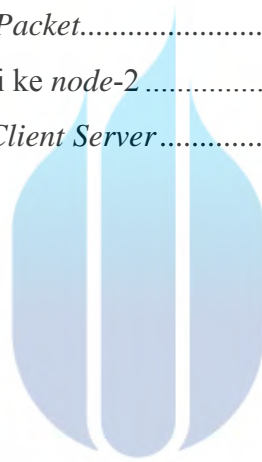
<b>ABSTRAK .....</b>	<b>II</b>
<b>ABSTRACT .....</b>	<b>III</b>
<b>PENGESAHAN TESIS.....</b>	<b>IV</b>
<b>PERNYATAAN SIMILARITY CHECK.....</b>	<b>V</b>
<b>PERNYATAAN.....</b>	<b>VI</b>
<b>KATA PENGANTAR.....</b>	<b>VII</b>
<b>DAFTAR ISI.....</b>	<b>VIII</b>
<b>DAFTAR GAMBAR.....</b>	<b>X</b>
<b>DAFTAR TABEL .....</b>	<b>XII</b>
<b>BAB I PENDAHULUAN</b>	
A. Latar Belakang.....	1
B. Rumusan Masalah .....	5
C. Tujuan Penelitian.....	6
D. Batasan Masalah.....	6
E. Kontribusi Penelitian .....	7
<b>BAB II TINJAUAN PUSTAKA</b>	
A. Literature Review .....	8
B. Quality of Service (QoS) .....	9
1) Throughput .....	10
2) Delay.....	10
3) Packet Loss.....	11
4) Jitter .....	11
C. Container dan Microservices .....	12
1) Docker .....	14
2) Kubernetes.....	16
D. Algoritma Load Balance.....	18
1) Round Robin.....	18
2) Least Connection .....	19

3) IP Hash .....	20
<b>BAB III METODOLOGI PENELITIAN</b>	
A. Tahapan Penelitian .....	22
B. Rancangan Topologi dan Environment Penelitian .....	25
C. Konfigurasi Environment Penelitian .....	26
D. Pengujian parameter .....	34
1) Skenario pengujian .....	35
<b>BAB IV HASIL DAN PEMBAHASAN</b>	
A. Pengujian skenario pertama .....	36
1) Hasil Pengujian CPU .....	37
2) Hasil Pengujian Memory .....	38
3) Hasil Pengujian Throughput .....	38
4) Hasil Pengukuran delay, packet loss, jitter .....	40
B. Pengujian skenario kedua .....	42
1) Hasil Pengujian CPU .....	42
2) Hasil Pengujian Throughput .....	43
3) Hasil Pengujian Memory .....	45
4) Hasil Pengujian delay, packet loss, jitter .....	46
C. Pengujian Skenario Ketiga .....	48
D. Sintesa .....	51
<b>BAB V KESIMPULAN DAN SARAN</b>	
A. Kesimpulan .....	54
B. Saran .....	55
<b>DAFTAR PUSTAKA .....</b>	<b>56</b>
<b>LAMPIRAN .....</b>	<b>60</b>

## DAFTAR GAMBAR

Gambar 2.1 Arsitektur VM vs <i>Container</i> .....	13
Gambar 2.2 <i>Docker</i> Architecture .....	25
Gambar 2.3. <i>Kubernetes</i> Architecture.....	17
Gambar 2.4 <i>Round Robin Flow</i> .....	19
Gambar 2.5 <i>Least Connection Flow</i> .....	20
Gambar 2.6 Proses <i>IP Hash</i> .....	21
Gambar 3.1 Flowchart Penelitian.....	22
Gambar 3.2 Topologi Penelitian .....	25
Gambar 3.3 Create <i>master</i> token.....	26
Gambar 3.4 <i>Kubectl get nodes</i> .....	27
Gambar 3.5 Mengaktifkan <i>Kubernetes</i> networking.....	27
Gambar 3.6 Join <i>Cluster</i> .....	28
Gambar 3.7 Konfigurasi <i>Nginx yml file</i> .....	28
Gambar 3.8 Pods Status .....	29
Gambar 3.9 <i>NodePort deployment</i> .....	29
Gambar 3.10 Verifikasi <i>NodePort</i> .....	30
Gambar 3.11 Konfigurasi <i>file haproxy.cfg</i> .....	30
Gambar 3.12 <i>Flowchart Round Robin</i> .....	31
Gambar 3.13 <i>Least Connection Diagram</i> .....	32
Gambar 3.14 <i>IP Hash Diagram</i> .....	34
Gambar 3.15 Apache Jmeter + wireshark Workflow.....	34
Gambar 4.1 CPU <i>Usage</i> Skenario Pertama.....	37
Gambar 4.2 Penggunaan <i>Memory</i> .....	38
Gambar 4.3 Total <i>packet (per second)</i> .....	39
Gambar 4.4 <i>Throughput</i> .....	40
Gambar 4.5 Statistik <i>wireshark IP Hash skenario pertama</i> .....	40
Gambar 4.6 Wireshark Round Robin Skenario Pertama .....	41
Gambar 4.7 Wireshark Least Connection Skenario Pertama.....	42
Gambar 4.8 Grafik Perbandingan Penggunaan CPU Skenario kedua .....	43

Gambar 4.9 Grafik Perbandingan antar Algoritma dalam Mengelola Packet .....	44
Gambar 4.10 Grafik perbandingan Throughput Skenario Kedua .....	45
Gambar 4.11 Penggunaan <i>Memory</i> Skenario Kedua .....	46
Gambar 4.12 <i>Wireshark Round robin</i> Skenario kedua .....	46
Gambar 4.13 <i>Wireshark Least connection</i> Skenario Kedua .....	47
Gambar 4.14 <i>Wireshark IP Hash</i> Skenario Kedua.....	48
Gambar 4.15 Uptime <i>Worker1</i> .....	48
Gambar 4.16 <i>Nginx services</i> .....	49
Gambar 4.17 <i>Node Status</i> .....	49
Gambar 4.18 Re-transmisi <i>Packet</i> .....	50
Gambar 4.19 Re-Transmisi ke <i>node-2</i> .....	50
Gambar 4.20 Komunikasi <i>Client Server</i> .....	51



UNIVERSITAS  
MERCU BUANA

## DAFTAR TABEL

Tabel 2.1 Standarisasi <i>Throughput</i> .....	10
Tabel 2.2 Standarisasi <i>Delay (Latency)</i> .....	10
Tabel 2.3 Standarisasi <i>Packet Loss</i> .....	11
Tabel 2.4 Standarisasi <i>Jitter</i> .....	12
Tabel 3.1 <i>Tools</i> pendukung penelitian .....	25
Tabel 3.2 Pengalamatan IP .....	26
Tabel 4.1 Total Request Waktu 5 Menit .....	36
Tabel 4.2 Nilai <i>Delay</i> dan <i>Jitter IP Hash Skenario Pertama</i> .....	41
Tabel 4.3 Nilai <i>Delay</i> dan <i>Jitter Round Robin Skenario Pertama</i> .....	41
Tabel 4.4 Nilai <i>Delay</i> dan <i>Jitter Least Connection Skenario Pertama</i> .....	42
Tabel 4.5 Nilai <i>Delay</i> dan <i>Jitter Round robin Skenario Kedua</i> .....	47
Tabel 4.6 Nilai <i>Delay</i> dan <i>Jitter Least connection Skenario Kedua</i> .....	47
Tabel 4.7 Nilai <i>Delay</i> dan <i>Jitter IP Hash Skenario Kedua</i> .....	48
Tabel 4.8 Hasil Pengujian Mengacu TIPHON .....	51