

**ANALISIS PERFORMA WEB E-COMMERCE DISTRO DENGAN
ALGORITMA ROUND ROBIN DAN DYNAMIC TIME SLICE
BERDASARKAN IPVSADM**

Skripsi



Disusun oleh:

PETRUS CHANEL REKSY ARFILA

71170234

PROGRAM STUDI INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA

2020

**ANALISIS PERFORMA WEB E-COMMERCE DISTRO
DENGAN ALGORITMA ROUND ROBIN DAN DYNAMIC
TIME SLICE BERDASARKAN IPVSADM**

Skripsi



Diajukan kepada Fakultas Teknologi Informasi Program Studi Informatika
Universitas Kristen Duta Wacana
Sebagai salah satu syarat dalam memperoleh gelar Sarjana Komputer

Diajukan oleh:

**PETRUS CHANEL REKSY ARFILA
71170234**

PROGRAM STUDI INFORMATIKA FAKULTAS TEKNOLOGI
INFORMASI

UNIVERSITAS KRISTEN DUTA WACANA

2020

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
SKRIPSI/TESIS/DISERTASI UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika Universitas Kristen Duta Wacana, saya yang bertanda tangan di bawah ini:

Nama : Petrus Chanel Reksy Arfila
NIM : 71170234
Program studi : Informatika
Fakultas : Teknologi Informasi
Jenis Karya : Skripsi/Tesis/Disertasi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Kristen Duta Wacana **Hak Bebas Royalti Noneksklusif** (*None-exclusive Royalty Free Right*) atas karya ilmiah saya yang berjudul:

“ANALISIS PERFORMA WEB E-COMMERCE DISTRO DENGAN ALGORITMA ROUND ROBIN DAN DYNAMIC TIME SLICE BERDASARKAN IPVSADM”

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti/Noneksklusif ini Universitas Kristen Duta Wacana berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama kami sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Yogyakarta
Pada Tanggal : 9 Agustus 2021

Yang menyatakan



(Petrus Chanel Reksy Arfila)

NIM.71170234

HALAMAN PERSETUJUAN

Judul Skripsi : ANALISIS PERFORMA WEB E-COMMERCE
DISTRO DENGAN ALGORITMA ROUND ROBIN
DAN DYNAMIC TIME SLICE BERDASARKAN
IPVSADM

Nama Mahasiswa : Petrus Chanel Reksy Arfila

N I M : 71170234

Matakuliah : Skripsi

Kode : TI0366

Semester : Genap

Tahun Akademik : 2020/2021

Telah diperiksa dan disetujui
Di Yogyakarta,
Pada tanggal 9 Juli 2021


Dosen Pembimbing I


Willy Sudiarto
Raharjo

Digitally signed by: Willy Sudiarto
Raharjo
Reason: I am approving this
document
Email: willysr@staff.ukdw.ac.id
Date: 2021/07/12 17:32:45 +07'0

Willy Sudiarto Raharjo, S.Kom., M.Cs .

Dosen Pembimbing II


Digitally signed by
12 Jul 2021

Digitally signed by Laurentius
Kuncoro Probo Saputra
DN: cn=Laurentius Kuncoro Probo
Saputra, o=Fakultas Teknologi
Informasi, ou=Informatika,
email=kuncoro@ti.ukdw.ac.id, c=ID
Date: 2021.07.12 12:18:11 +07'00'

Laurentius Kuncoro P, ST., M. Eng .

HALAMAN PENGESAHAN
ANALISIS PERFORMA WEB E-COMMERCE DISTRO DENGAN
ALGORITMA ROUND ROBIN DAN DYNAMIC TIME SLICE
BERDASARKAN IPVSADM

Oleh: 71170234 / Petrus Chanel Reksy Arfila

Dipertahankan di depan Dewan Penguji Skripsi

Program Studi Informatika Fakultas Teknologi Informasi

Universitas Kristen Duta Wacana – Yogyakarta

Dan dinyatakan diterima untuk memenuhi salah satu syarat memperoleh gelar

Sarjana Komputer

Pada tanggal 15 Juli 2021

Yogyakarta, 23 Juni 2021

Mengesahkan,

Dewan Penguji :

1. Willy Sudiarto Raharjo, S.Kom., M.Cs

2. Laurentius Kuncoro Probo Saputra, S.T., M.Eng

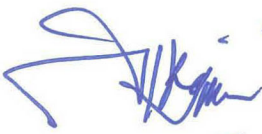
Digitally signed:
12 Jul 2021
3. Joko Purwadi, S.Kom., M.Kom

4. Aditya Wikan Mahastama, S.Kom., M.Cs


Dekan,


Restyandito, S.Kom., MSiS.,
Ph.D.


Ketua Program Studi


Gloria Virginia, S.Kom., MAI.,
Ph.D.

PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

ANALISIS PERFORMA WEB E-COMMERCE DISTRO DENGAN ALGORITMA ROUND ROBIN DAN DYNAMIC TIME SLICE BERDASARKAN IPVSADM

yang saya kerjakan untuk melengkapi sebagian persyaratan menjadi Sarjana Komputer pada pendidikan Sarjana Program Studi Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana, bukan merupakan tiruan atau duplikasi dari skripsi kesarjanaan di lingkungan Universitas Kristen Duta Wacana maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Jika dikemudian hari didapati bahwa hasil skripsi ini adalah hasil plagiasi atau tiruan dari skripsi lain, saya bersedia dikenai sanksi yakni pencabutan gelar kesarjanaan saya.

Yogyakarta, 27 Mei 2021



PETRUS CHANEL REKSY ARFILA
71170234

UCAPAN TERIMA KASIH

Puji syukur senantiasa penulis panjatkan kepada Tuhan Yang Maha Esa yang telah memberikan kesehatan dan kekuatan dalam penyusunan skripsi ini. Laporan ini dibuat untuk memenuhi persyaratan tugas Mata Kuliah Skripsi di Program Studi Informatika, Fakultas Teknologi Informasi Universitas Kristen Duta Wacana Yogyakarta. Tujuan dibuatnya laporan skripsi ini yaitu untuk menjabarkan proses penelitian yang dilakukan oleh penulis dari awal hingga akhir.

Dalam proses penyusunan laporan skripsi, tentu tidak lepas dari pengarahan dan bimbingan oleh berbagai pihak. Oleh karena itu penulis mengucapkan terima kasih sebesar-besarnya kepada semua pihak yang membantu, di antaranya sebagai berikut:

1. Bapak Willy Sudiarto Raharjo, S.Kom., M.Cs dan Bapak Laurentius Kuncoro P, ST., M. Eng. selaku dosen pembimbing Mata Kuliah Skripsi.
2. Bapak Restyandito, S.Kom, MSIS., Ph.D. selaku Dekan Fakultas Teknologi Informasi Universitas Kristen Duta Wacana.
3. Ibu Gloria Virginia, Ph.D. selaku Kepala Program Studi Informatika Universitas Kristen Duta Wacana.
4. Orang tua dan keluarga yang telah memberi doa serta dukungan sehingga proses penelitian hingga penyusunan laporan dapat berjalan dengan lancar.

Penulis sadar bahwa laporan skripsi yang disusun masih jauh dari kesempurnaan. Oleh karena itu penulis akan menghargai segala bentuk kritik dan saran yang bersifat membangun. Penulis berharap laporan ini dapat memiliki nilai positif serta nilai manfaat bagi semua pihak yang membutuhk

INTISARI

Mudahnya dalam bertransaksi di internet , telah meningkatkan jumlah pengguna yang menggunakan toko online sebagai tempat mereka berbelanja. Aktivitas bertransaksi online yang banyak ini jika tidak dimanajemen dengan baik akan membuat server menjadi kelebihan beban (overload) karena tidak dapat menampung request data yang diterima. Agar tidak terjadi overload pada server.

Pada penelitian ini mencoba untuk menerapkan teknik load balancing dengan algoritma round robin pada website toko online. Pengujian ini dilakukan dengan cara menggunakan tools load testing sebanyak 50 hingga 100 request yang kemudian mencatat hasil dari “http_req_duration”, “http_req_receiving” dan juga mencatat resource CPU dan RAM yang digunakan oleh server saat memproses data .

Hasil pada penelitian ini adalah *loadbalancer Openlitespeed* lebih optimal dalam melayani request *read* dan *write*, manajemen *resource CPU* dan *RAM* yang digunakan pada Loadbalancer dibagi rata di antara kedua server yang membuat segala request yang diminta jadi lebih cepat untuk di terima oleh client.

Kata kunci: Load Balancing, Round Robin, Web Server, E-Commerce, Openlitespeed, Nginx

ABSTRACT

The ease of transacting on the internet has increased the number of users who use online stores as a place to shop. This many online transaction activities if not managed properly will make the server overload because it cannot accommodate the received data requests. In order to avoid overload on the server. In this study, we try to apply a load balancing technique with a round robin algorithm on an online store website. This test is done by using load testing tools of 50 to 100 requests which then records the results of "http_req_duration", "http_req_receiving" and also records CPU and RAM resources used by the server when processing data. The results of this study are Openlitespeed loadbalancer is more optimal in serving read and write requests, CPU and RAM resource management used in Loadbalancer is divided equally between the two servers which makes all requests requested faster to be received by the client.

Keywords: Load Balancing, Round Robin, Web Server, E-Commerce,Openlitespeed,nginx

DAFTAR ISI

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS SECARA ONLINE.....	III
HALAMAN PERSETUJUAN.....	IV
HALAMAN PENGESAHAN	V
UCAPAN TERIMA KASIH.....	VI
INTISARI	VII
DAFTAR ISI.....	VIII
DAFTAR GAMBAR	X
DAFTAR TABEL.....	XII
BAB 1	1
PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
BAB 2	4
TINJAUAN PUSTAKA	4
2.1 Tinjauan Pustaka.....	4
2.2 Landasan teori.....	5
2.2.1 Algoritma Round Robin.....	5
2.2.2 Webserver	7
2.2.3 Nginx.....	7
2.2.4 Openlitespeed.....	8
2.2.5 Load Balancing	8
2.2.6 AWS (Amazon Web Service).....	9
BAB 3	10
METODOLOGI DAN PERANCANGAN SISTEM.....	10
3.1 Spesifikasi Sistem	10
3.1.1 Fungsional.....	10
3.1.2 Non-Fungsional.....	10
3.2 Rancangan Penelitian.....	11
3.2.1 Blok Diagram Sistem.....	11
3.2.2 Proses instalasi Software.....	11
3.2.3 Konfigurasi Loadbalancer pada Nginx	13
3.2.4 Konfigurasi Loadbalancer pada Openlitespeed	14

3.3 Metodologi Penelitian.....	15
BAB IV	19
PENGAMATAN DAN ANALISIS PENELITIAN	19
4.1 Pengamatan.....	19
4.1.1 Pengamatan pada Web Server Nginx	19
4.1.1.1 Dengan Loadbalancer	19
4.1.1.2 Tanpa Loadblancer Server 1	23
4.1.1.3 Tanpa Loadblancer Server 2	27
4.1.2 Pengamatan pada Web Server Openlitespeed.....	30
4.1.2.1 Dengan Loadbalancer	30
4.1.2.2 Tanpa Loadblancer Server 1	35
4.1.2.3 Tanpa Loadblancer Server 2	38
4.2 Analisis Penelitian	40
BAB V	44
KESIMPULAN DAN SARAN.....	44
5.1 Kesimpulan	44
5.2 Saran	44
DAFTAR PUSTAKA	45
LAMPIRAN.....	46

© UKDW

DAFTAR GAMBAR

Gambar 2.1. Flowchart Algoritma Round Robin.....	6
Gambar 3.1 . Blok Diagram Sistem.....	11
Gambar 3.2. Daftar security groups.....	11
Gambar 3.3. Inbound rules pada website security groups.....	12
Gambar 3.4. Inbound rules pada Database security groups.....	12
Gambar 3.5 Konfigurasi Loadbalancer Nginx.....	13
Gambar 3.6 Konfigurasi Loadbalancer Openlitespeed.....	14
Gambar 3.7 Konfigurasi Loadbalancer Openlitespeed server 1.....	14
Gambar 3.8 Konfigurasi Loadbalancer Openlitespeed server 2.....	14
Gambar 3.9 Konfigurasi Loadbalancer Openlitespeed Loadbalancer.....	14
Gambar 3.10 . Contoh Script pengujian K6.....	16
Gambar 3.11 . Contoh Hasil Netdata.....	17
Gambar 3.12 . Contoh Hasil Netdata bagian CPU usage.....	17
Gambar 3.13 . Contoh Hasil Netdata bagian RAM Usage.....	17
Gambar 3.14 . Contoh Selenium IDE dengan 20 user mengakses ke website.....	18
Gambar 4.1 Hasil Pengujian K6 Loadbalancer Nginx.....	19
Gambar 4.2 Hasil Penggunaan CPU pada Server Loadbalancer Nginx.....	20
Gambar 4.3 Hasil Penggunaan RAM pada Server Loadbalancer Nginx.....	20
Gambar 4.4 Hasil Penggunaan CPU pada Server 1 dengan Loadbalancer Nginx.....	20
Gambar 4.5 Hasil Penggunaan RAM pada Server 1 dengan Loadbalancer Nginx.....	20
Gambar 4.6 Hasil Penggunaan CPU pada Server 2 dengan Loadbalancer Nginx.....	20
Gambar 4.7 Hasil Penggunaan RAM pada Server 2 dengan Loadbalancer Nginx.....	20
Gambar 4.8 Hasil Pengujian Write dengan aplikasi Selenium Loadbalancer Nginx.....	21
Gambar 4.9 Hasil Pengujian K6 server 1 Nginx tanpa Loadbalancer.....	23
Gambar 4.10 Hasil penggunaan CPU pada Server 1 Nginx tanpa Loadbalancer.....	23
Gambar 4.11 Hasil penggunaan RAM pada Server 1 Nginx tanpa Loadbalancer.....	23
Gambar 4.12 Hasil Pengujian Write dengan aplikasi Selenium server 1.....	24
Gambar 4.13 Pengujian K6 Server 2 tanpa Loadbalancer Nginx.....	25
Gambar 4.14 Hasil Penggunaan CPU pada Server 2 tanpa Loadbalancer Nginx.....	26
Gambar 4.15 Hasil Penggunaan RAM pada Server 2 tanpa Loadbalancer Nginx.....	26
Gambar 4.16 Hasil Pengujian Write dengan aplikasi Selenium server 2.....	26
Gambar 4.17 Hasil Pengujian K6 Loadbalancer Openlitespeed.....	28

Gambar 4.18 Hasil Penggunaan CPU pada Server Loadbalancer Openlitespeed.....	29
Gambar 4.19 Hasil Penggunaan RAM pada Server Loadbalancer Openlitespeed.....	29
Gambar 4.20 Hasil Penggunaan CPU pada Server 1 Loadbalancer Openlitespeed.....	29
Gambar 4.21 Hasil Penggunaan RAM pada Server 1 Loadbalancer Openlitespeed.....	29
Gambar 4.22 Hasil Penggunaan CPU pada Server 2 Loadbalancer Openlitespeed.....	29
Gambar 4.23 Hasil Penggunaan RAM pada Server 2 Loadbalancer Openlitespeed.....	29
Gambar 4.24 Hasil Pengujian Write dengan aplikasi Selenium server Loadbalancer.....	30
Gambar 4.25 Hasil Pengujian K6 server 1 Openlitespeed tanpa Loadbalancer.....	32
Gambar 4.26 Hasil Penggunaan CPU pada Server 1 Openlitespeed tanpa Loadbalancer....	32
Gambar 4.27 Hasil Penggunaan RAM pada Server 1 Openlitespeed tanpa Loadbalancer...32	32
Gambar 4.28 Hasil Pengujian Write dengan aplikasi Selenium server 1.....	33
Gambar 4.29 Hasil Pengujian K6 server 2 Openlitespeed tanpa Loadbalancer.....	35
Gambar 4.30 Hasil Penggunaan CPU pada Server 2 Openlitespeed tanpa Loadbalancer....35	35
Gambar 4.31 Hasil Penggunaan RAM pada Server 2 Openlitespeed tanpa Loadbalancer...35	35
Gambar 4.32 Hasil Pengujian Write dengan aplikasi Selenium server 2.....	36
Gambar 4.33 Perbandingan hasil rata-rata pengujian Loadbalancer Nginx dan Openlitespeed.....	38
Gambar 4.34 Perbandingan rata-rata webserver Nginx dan Openlitespeed pada server 1 tanpa Loadbalancer.....	38
Gambar 4.35 Perbandingan rata-rata webserver Nginx dan Openlitespeed pada server 2 tanpa Loadbalancer.....	39

DAFTAR TABEL

Tabel 4.1 Hasil Pengujian Loadbalancer Nginx	22
Tabel 4.2 Hasil Pengujian Server 1 Nginx tanpa Loadbalancer	24
Tabel 4.3 Hasil Pengujian Server 2 Nginx tanpa Loadbalancer	27
Tabel 4.4 Hasil Pengujian Server Loadbalancer Openlitespeed	31
Tabel 4.5 Hasil Pengujian Server 1 Openlitespeed tanpa Loadbalancer	34

©UKDW

BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Komputasi awan menjadi semakin populer dimana sumber daya komputasi tersedia sesuai permintaan pengguna dan sesuai dengan kebutuhan. Komputasi awan menciptakan peluang baru untuk bidang TI dan bisnis. Komputasi awan menggunakan teknologi internet untuk penyampaian kapabilitas yang Mendukung setiap layanan pengguna yang dibutuhkan yaitu melalui komputasi awan kita dapat mengakses layanan apapun yang kita inginkan dari mana saja ke komputer tanpa mengkhawatirkan hal seperti penyimpanan, biaya, pemeliharaan system, dan sebagainya.

Di dunia industri banyak perusahaan menggunakan website sebagai media promosi dan transaksi online seperti e-commerce. Untuk menangani jutaan request ke website ada banyak hal yang dapat menyebabkan kegagalan dalam pembuatan web salah satunya adalah banyaknya permintaan dan transaksi ke website yang dapat memperlambat system sehingga transaksi tidak dapat dilakukan yang mengakibatkan kerugian bagi pengusaha kecil.

Hosting berbayar adalah cara untuk menghadapi kendala tersebut tetapi bila untuk pengusaha kecil tentu akan memberikan masalah yang besar dikarenakan biaya yang mahal untuk menyewa server hosting. Dari permasalahan itu penulis ingin menggunakan AWS sebagai server dan menggunakan web server yang open source untuk menekan biaya dan juga menerapkan teknologi Load Balancing jaringan.

Untuk penggunaan server di sini menggunakan layanan cloud dari AWS, AWS adalah layanan server dengan opsi yang terjangkau oleh pengusaha jika ingin membuat e-commerce. AWS adalah penyedia layanan cloud computing, dimana setiap fungsi dapat diakses dengan menggunakan Web Service. Konsep dalam Amazon Web Service (AWS) berupa instance. dengan menggunakan layanan virtual, para pengguna dapat melakukan pengembangan aplikasi di banyak platform. disisi lain AWS juga menyediakan layanan cloud yang terjangkau serta kebebasan dalam kostumisasi server jika digunakan oleh pengusaha kecil.

Untuk optimalisasi server disini penulis menggunakan Load Balancing. Load balancing merupakan teknik pendistribusian trafik pada proses yang berbeda sehingga meningkatkan kemampuan dalam pemrosesan pada request yang masuk. Loadbalancer membagi request sehingga terbentuk antrian. Dengan menggunakan Loadbalancer mempunyai keuntungan

menurunkan beban pemrosesan data yang diolah oleh Server sebagai penerima, sehingga Server akan menangani permintaan dan kinerjanya akan stabil dikarenakan berkurangnya proses yang diperlukan Server untuk memproses data.

Kinerja pada web server ini menggunakan Algoritma Round Robin agar dapat memproses request dari client dengan cepat tanpa membebani CPU . Algoritma Round Robin merupakan algoritma yang digunakan oleh penjadwal proses dan jaringan dalam komputasi. Algoritma ini bekerja dengan membagi beban proses secara urut dari satu proses ke proses lain seperti perulangan. Algoritma ini mengolah data dan mengurutkannya secara bergantian. dari algoritma tersebut akan dapat dilihat web server mana yang terbaik untuk digunakan untuk e-commerce.

Web Server yang akan digunakan untuk menjadi perbandingan adalah Nginx dan LiteSpeed. Web Server adalah perangkat lunak yang berguna untuk menjalankan situs web. request dikirim menggunakan Hypertext Transfer Protocol (HTTP). Web server mengirimkan hasil request client yang berupa dokumen HTML, gambar, css, tes, dll. dan kemudian akan ditampilkan melalui Peramban Web.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang ditulis sebelumnya dapat disimpulkan rumusan masalah sebagai berikut:

1. Web Server apa yang sesuai untuk e-commerce kecil jika dilihat dari performanya?
2. Bagaimana performa Web Server sebelum dan sesudah penerapan Loadbalancer?
3. Bagaimana hasil performa dari Load Balancing dengan algoritma Round Robin pada Web Server dengan parameter `http_req_duration`, `http_req_receiving`, penggunaan CPU dan RAM?

1.3 Batasan Masalah

Untuk fokus penelitian, penulis memberikan batasan batasan penelitian, yaitu :

1. Sulitnya mendapatkan 100 user di waktu yang sama maka 100 user tersebut diganti menjadi bot.
2. Fokus Pengukuran hanya pada performansi web server berdasarkan `http_req_duration`, `http_req_receiving`, penggunaan CPU dan RAM
3. Web server yang digunakan menggunakan Nginx dan Openlitespeed.

1.4 Tujuan Penelitian

Tujuan dari penelitian yang akan dilaksanakan dapat dirumuskan sebagai berikut:

1. untuk mengetahui load Web Server sebelum dan sesudah penggunaan Loadbalancer.
2. untuk mengetahui Web Server apa yang tepat diantara Nginx dan Openlitespeed untuk digunakan di e-commerce.

1.5 Manfaat Penelitian

Diharapkan dengan melakukan penelitian ini maka dapat didapatkan manfaat:

1. Bagi Peneliti : Memperbarui ilmu dan wawasan dalam hal *Administrasi System* .
2. Manfaat Teoritis : Hasil penelitian dapat dijadikan bahan referensi bagi penelitian selanjutnya maupun penelitian lain.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan pengujian dan pengamatan pada bab 4, dapat diperoleh hasil:

1. Dari hasil pengujian sebanyak 2 kali, kesimpulan yang dapat diambil adalah *loadbalancer Openlitespeed* lebih optimal dalam melayani request *read* dan *write*, manajemen *resource CPU* dan *RAM* yang digunakan pada Loadbalancer dibagi rata di antara kedua server yang membuat segala request yang diminta jadi lebih cepat untuk di terima oleh client. Dibuktikan dari hasil *http_req_duration*, *http_req_receiving*, dan penggunaan *CPU dan RAM* yang digunakan untuk menjalankan test yang diberikan lebih cepat dan lebih optimal jika dibandingkan dengan *Loadbalancer Nginx*.
2. Dari pengujian tanpa menggunakan Loadbalancer yang di lakukan sebanyak 2 kali, kesimpulan yang dapat diambil adalah sebuah website toko online tanpa menggunakan Loadbalancer jika di akses dengan operasi *write* dan *read* secara bersamaan dapat membuat website menjadi down. Hal tersebut dikarenakan server menggunakan semua resource untuk menangani setiap request dari client, sehingga ada beberapa data yang tidak dapat di selesaikan karena server terlalu banyak memproses data yang di inginkan client.

5.2 Saran

Adapun saran yang penulis dapat berikan untuk pengembangan penelitian selanjutnya antara lain:

1. Gunakanlah akun AWS berbayar dan tidak menggunakan akun AWS Educate dikarenakan minimnya fitur AWS yang dapat digunakan.
2. Dapat dikembangkan menggunakan server yang dikonfigurasi pada docker atau platform cloud lainnya.
3. Menggunakan Benchmark Tools yg lain selain K6 dan Selenium.
4. Jangan melakukan load testing berlebihan karena bisa mengakibatkan akun AWS menjadi disuspend.

Daftar Pustaka

- Aziz, M., Fuad, A. and Jamil, M., 2018. IMPLEMENTASI CLOUD COMPUTING SEBAGAI INFRASTRUKTUR LAYANAN MAIL SERVER PADA UNIVERSITAS KHAIRUN. *JIKO (Jurnal Informatika dan Komputer)*, 1(1), pp.43-50.
- Komaruddin, A., Sipitorini, D. and Rispian, P., 2019. Load Balancing with Round Robin method for sharing of Web Server work load. *Jurnal Siliwangi*, [online] 5(2), pp.47-50. Available at:<<http://jurnal.unsil.ac.id/index.php/jssainstek/article/view/1184>>[Accessed 29 November 2020].
- Kumar,G. (2013). Design an Embedded Web Server for Monitoring and Controlling Systems or Devices. *Design An Embedded Web Server For Monitoring And Controlling Systems Or Devices*.Kurniawan, Y., Sabriasyah & Sakti, E., 2013. Analisis Kinerja Algoritma Load Balancer dan Implementasi pada Layanan Web. Universitas Brawijaya, Volume III.
- Luthfi,M., Data, M., & Yahya, W. (2017). Perbandingan Performa Reverse Proxy Caching Nginx dan Varnish Pada Web Server Apache. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 2(4), 1457-1463
- Mahmood, A. & Rashid, I., 2011. Comparison of Load Balancing Algorithms for Cluster Web Server. Malaysia, IEEE. ISBN : 978-1-4577-0988-3.
- Mbayo Kabango, C. and Romeo Asa, A., 2015.Factors influencing e-commerce development:Implications for the developing countries. *International Journal Of Innovation And Economic Development*,1(1), pp.59-66.
- Muhammad, M., Data, M. and Yahya, W.,2018. Perbandingan Performa Reverse ProxyCaching Nginx dan Varnish Pada WebServer Apache. *Jurnal PengembanganTeknologi Informasi dan Ilmu Komputer*,2, pp.1457-1463.
- Nugroho, A. and Mustofa, K., 2012. *Implementasi Komputasi Awan Menggunakan TeknologiGoogle App Engine (GAE) dan Amazon Web Services*.Volume 6Rabu, J.A., Purwadi, J. & Raharjo, W.S., 2012.
- Implementasi Load Balacing Web Server Menggunakan Metode LVS-NAT. *INFORMATIKA*.Volume 8 : 2.Singh, H. & Kumar, D.S., 2011. Dispatcher Based Dynamic Load Balancing on Web Server System. *International Journal of Grid and Distributed Computing*. ISSN : 2160-9772.
- Siva Nageswara Rao, G., Srinivasu, N.,Srinivasu, S. and Koteswara Rao, G., 2015. Dynamic Time Slice Calculation forRound Robin Process Scheduling Using NOC. *International Journal of Electrical and Computer Engineering (IJECE)*, 5(6), p.1480.
- Yusuf, E., A Riza, T., & Ariefianto, T. (2013). IMPLEMENTASI TEKNOLOGI LOAD BALANCER DENGAN WEB SERVER NGINX UNTUK MENGATASI BEBAN SERVER. *Seminar Nasional Teknologi Informasi Dan Multimedia 2013*, 01(01).