

Load Balancing Menggunakan Algoritma Round Robin Dengan Stickiness Pada AWS

Meidiana Shafira[#], Hidra Amnur[#], Rita Afyenni[#]

[#]*Jurusan Teknologi Informasi, Politeknik Negeri Padang, Limau Manis, Padang, 25164, Indonesia*
E-mail: hidraamnur@gmail.com, ritaafyenni@pnp.ac.id

ABSTRACTS

The development of information and communication technology is increasingly sophisticated, especially in the field of networks. This development is closely related to the internet which is one of the most popular information providers today. The increasing need for information will demand fast access to get the latest information, one of which is the most dominant influencing the speed of access to a particular website is the service provider. Therefore it takes a server that has good quality and performance or has many servers so that it is not burdened on one server. Application load balancing is one of the right solutions to overcome this problem. The load balancing application will share the network load and avoid overloading one of the servers. The load balancing application method used is round robin with stickiness. When one server crashes, the load balancing application will automatically redirect requests to another server that is still functioning, so that the website can be accessed.

KATA KUNCI

*Load Balancing,
Web Services,
Round Robin,
Stickiness,*

ABSTRAK

Perkembangan teknologi informasi dan komunikasi semakin lama semakin canggih terutama dibidang jaringan. Perkembangan tersebut berkaitan erat dengan internet yang merupakan salah satu penyedia informasi terpopuler saat ini. Meningkatnya kebutuhan akan informasi menuntut akses yang cepat untuk mendapatkan informasi-informasi terkini, salah satunya yang paling dominan mempengaruhi kecepatan akses suatu alamat website tertentu adalah server penyedia layanan. Maka dari itu dibutuhkan server yang mempunyai kualitas dan performa yang bagus atau memiliki banyak server sehingga tidak terbebani pada satu server. Application load balancing merupakan merupakan salah satu solusi yang tepat untuk mengatasi masalah tersebut. Application load balancing akan membagi beban jaringan dan menghindari pembebanan berlebih pada salah satu server. Metode Application load balancing yang digunakan yaitu round robin dengan stickiness. Ketika salah satu server mengalami kerusakan maka Application load balancing akan otomatis mengarahkan request ke server lain yang masih berfungsi, sehingga website masih bisa diakses.

1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi semakin lama semakin canggih, terutama di bidang jaringan. Perkembangan tersebut berkaitan erat dengan internet yang merupakan salah satu penyedia informasi terpopuler saat ini. Meningkatnya kebutuhan akan informasi menuntut akses yang cepat untuk mendapatkan informasi-informasi terkini, salah satunya yang paling dominan mempengaruhi kecepatan akses suatu alamat website tertentu adalah server penyedia layanan.

Instansi-instansi maupun pengusaha memanfaatkan sistem informasi berupa website dalam melakukan semua urusan bisnisnya, memerlukan server yang handal dalam pemrosesan khususnya ketika website itu mulai banyak

pengunjung yang mengakses sehingga mengakibatkan web server menjadi sibuk. Hal ini akan mengurangi performa web server tersebut, sehingga mengakibatkan sulitnya pengaksesan informasi karena server sibuk. Jika server tidak bisa diakses dengan baik maka bisnis akan berhenti dan menyebabkan kerugian.

Server dengan spesifikasi tinggi sangat dibutuhkan untuk membangun web server dengan kualitas yang bagus, sehingga mampu memberikan layanan yang cepat bagi klien, namun biaya yang dibutuhkan sangat mahal dan masih sering terjadinya down, yang akan mengakibatkan server tidak bisa diakses.

Dengan menggabungkan beberapa server menjadi kesatuan yang dapat berkerja secara bersamaan untuk menyelesaikan masalah, dalam masalah ini pembagian beban server pada web server dengan menggunakan teknik Application load balancing sangat perlu untuk dilakukan untuk melakukan pemerataan beban yang bertujuan untuk meringankan beban yang ditanggung masing-masing server sehingga ketika salah satu server mengalami masalah dapat secara otomatis digantikan oleh server lain dan server masih tetap bisa diakses. Ada beberapa algoritma yang digunakan untuk load balancing salah satunya yaitu round robin dengan stickiness yang bisa dijadikan solusi dan jawaban yang tepat untuk mengatasi permasalahan di atas.

2. METODOLOGI PENELITIAN

2.1 Load balancing

Load balancing adalah suatu teknik yang digunakan untuk memisahkan antara dua atau banyak network link. Dengan mempunyai banyak link maka optimalisasi utilitas sumber daya, throughput atau response time akan semakin baik karena mempunyai lebih dari satu link yang bias saling mem-backup pada saat network down dan menjadi cepat pada saat network normal jika memerlukan realibilitas tinggi yang memerlukan 100% koneksi uptime dan yang menginginkan koneksi upstream yang berbeda dan dibuat saling mem-backup.

Load balancing merupakan salah satu mekanisme untuk membagi beban komputasi ke beberapa server. Load balancing bertujuan untuk mengoptimalkan sumber daya, memaksimalkan throughput, meminimalkan waktu respon, dan menghindari pembebanan berlebihan di satu sumber daya. Menggunakan beberapa sumber daya komputasi juga dapat mengurangi kemungkinan tidak berfungsinya suatu layanan karena setiap sumber daya dapat saling menggantikan (redundant).

Sistem Load balancing terdiri dari virtual server dan real server. Virtual server sekaligus berfungsi sebagai perangkat yang membagi beban ke server sebenarnya. Virtual Server akan dituju oleh akses client. Saat sebuah permintaan dari client diterima oleh virtual server, akan diteruskan ke server sesungguhnya berdasarkan data beban masing-masing server. Secara umum server yang dituju adalah server dengan beban paling sedikit. Mekanisme pemilihan server inilah yang terdiri dari beberapa mekanisme seperti[6]:

1. Least connection: mengutamakan real server dengan koneksi paling sedikit.
2. Round robin: menempatkan semua real server pada antrian yang melingkar dan mengalokasikan koneksi bergantian untuk setiap turn.
3. Destination hash: menggunakan data tabel statis dari alamat IP tujuan untuk mengalokasikan koneksi.
4. Source hash scheduling: menggunakan data tabel statis dari alamat IP asal (client) untuk mengalokasikan koneksi.

2.2 Round Robin

Round robin merupakan metode algoritma yang paling sederhana dan paling banyak digunakan oleh perangkat load balancing. Metode round robin bekerja dengan cara membagi beban secara bergiliran dan berurutan dari satu server ke server lainnya. Konsep dasar dari round robin ini adalah dengan menggunakan time sharing, pada intinya algoritma ini memproses antrian secara bergiliran.

Metode algoritma round robin bekerja dengan menanggapi permintaan tidak hanya dengan satu alamat IP, tetapi beberapa server yang memiliki host yang identik. Alamat IP yang mencoba melakukan koneksi pertama kali akan dikerjakan terlebih dahulu, sehingga pada saat melakukan koneksi yang berbeda, client akan menerima dari server yang berbeda pula, maka pendistribusian beban keseluruhan antar server dapat sama.[7]

2.3 Web Server

Web server merupakan sebuah software yang memberikan layanan data yang berfungsi untuk menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan browser web dan mengirimkan kembali hasilnya dalam bentuk halaman - halaman web yang umumnya berbentuk dokumen HTML. Beberapa aplikasi yang memiliki fungsi sebagai web server diantaranya, apache Tomcat, Microsoft windows Server 2003 Internet Information Services (IIS), Lighttpd, Sun Java System Web Server, Xitami Web Server, dan Zeus Web Server.[8]

2.4 Cloud computing

Komputasi awan (cloud computing) adalah gabungan pemanfaatan teknologi komputer ('komputasi') dan pengembangan berbasis Internet ('awan'). Awan (cloud) adalah metafora dari internet, sebagaimana awan yang sering digambarkan di diagram jaringan komputer. Sebagaimana awan dalam diagram jaringan komputer tersebut,

awan (cloud) dalam Cloud computing juga merupakan abstraksi dari infrastruktur kompleks yang disembunyikannya. Ia adalah suatu metoda komputasi di mana kapabilitas terkait teknologi informasi disajikan sebagai suatu layanan (as a service), sehingga pengguna dapat mengaksesnya lewat Internet ("di dalam awan") tanpa mengetahui apa yang ada didalamnya, ahli dengannya, atau memiliki kendali terhadap infrastruktur teknologi yang membantunya. Menurut sebuah makalah tahun 2008 yang dipublikasi IEEE Internet Computing "Cloud computing adalah suatu paradigma di mana informasi secara permanen tersimpan di server di internet dan tersimpan secara sementara di komputer pengguna (client) termasuk di dalamnya adalah desktop, komputer tablet, notebook, komputer tembok, handheld, sensor-sensor, monitor dan lain- lain."[9]

2.5 Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) menyediakan kapasitas komputasi yang dapat diskalakan di Amazon Web Layanan cloud (AWS). Menggunakan Amazon EC2 menghilangkan kebutuhan pengguna untuk berinvestasi di perangkat keras di awal, jadi pengguna dapat mengembangkan dan menerapkan aplikasi lebih cepat. pengguna dapat menggunakan Amazon EC2 untuk meluncurkan virtual sebanyak atau sesedikit server sesuai kebutuhan pengguna, konfigurasi keamanan dan jaringan, serta kelola penyimpanan. Amazon EC2 memungkinkan pengguna untuk meningkatkan atau menurunkan, untuk menangani perubahan dalam persyaratan atau lonjakan popularitas, mengurangi kebutuhan pengguna memperkirakan lalu lintas.[10]

Amazon EC2 menyediakan fitur-fitur berikut:

1. Lingkungan komputasi virtual, yang dikenal sebagai instance.
2. Templat yang telah dikonfigurasi sebelumnya untuk instance pengguna, yang dikenal sebagai Amazon Machine Images (AMI), paket itu bit yang dibutuhkan untuk server (termasuk sistem operasi dan perangkat lunak tambahan).
3. Berbagai konfigurasi CPU, memori, penyimpanan, dan kapasitas jaringan untuk instance.
4. Mengamankan informasi login untuk instance menggunakan key pair (AWS menyimpan kunci publik, dan menyimpannya kunci pribadi di tempat yang aman).
5. Volume penyimpanan untuk data sementara yang dihapus saat pengguna menghentikan atau menghentikan instance pengguna, diketahui sebagai contoh penyimpanan volume.
6. Volume penyimpanan persisten untuk data pengguna menggunakan Amazon Elastic Block Store (Amazon EBS), yang dikenal sebagai Volume Amazon EBS.
7. Beberapa lokasi fisik untuk sumber daya, seperti instance dan volume Amazon EBS, yang dikenal sebagai Region and Availability Zone.
8. Firewall yang memungkinkan pengguna menentukan protokol, port, dan rentang IP sumber yang dapat menjangkau pengguna contoh menggunakan security group.
9. Alamat IPv4 statis untuk komputasi awan dinamis, yang dikenal sebagai Elastic IP address.
10. Metadata, yang dikenal sebagai tag, yang dapat dibuat dan ditetapkan ke sumber daya Amazon EC2.
11. Jaringan virtual yang dapat dibuat yang diisolasi secara logis dari sisa cloud AWS, dan dapat secara opsional terhubung ke jaringan sendiri, yang dikenal sebagai virtual private cloud (VPC)

2.6 Amazon RDS

Amazon Relational Database Service (Amazon RDS) adalah layanan web yang mempermudah penyetaan, mengoperasikan, dan menyekalakan database relasional di AWS Cloud. Ini menyediakan kapasitas yang hemat biaya dan dapat diubah ukurannya untuk database relasional standar industri dan mengelola tugas administrasi database umum.[11]

Database engine yang ada pada Amazon RDS:

1. MariaDB
2. Microsoft SQL Server
3. MySQL
4. Amazon Aurora
5. Oracle
6. PostgreSQL

2.7 Elastic IP

Elastic IP address adalah alamat IPv4 statis yang dirancang untuk komputasi awan dinamis. Dengan menggunakan Elastic IP address, pengguna dapat menutupi kegagalan instance atau perangkat lunak dengan memetakan ulang alamat secara cepat ke instance lain di akun pengguna. Elastic IP address dialokasikan ke akun AWS, dan menjadi milik pengguna sampai pengguna melepaskannya.

Elastic IP address adalah alamat IPv4 publik, yang dapat dijangkau dari internet. Jika instance pengguna tidak memiliki alamat IPv4 publik, pengguna dapat mengaitkan Elastic IP address dengan instance untuk mengaktifkan komunikasi dengan internet. Misalnya, ini memungkinkan pengguna untuk terhubung ke instance dari komputer lokal.[10]

2.8 Apache

Apache merupakan web server yang paling banyak dipergunakan di Internet. Program ini pertama kali didesain untuk sistem operasi lingkungan UNIX, untuk saat ini telah tersedia apache yang di desain untuk sistem operasi lainnya. Apache mempunyai program pendukung yang cukup banyak. Hal ini memberikan layanan yang cukup lengkap bagi penggunaanya. [12]

2.9 Wordpress

Wordpress adalah sebuah aplikasi sumber terbuka (open source) yang sangat populer digunakan sebagai mesin blog (blog engine). Wordpress dibangun dengan bahasa pemrograman PHP dan basis data (database) MySQL. PHP dan MySQL, keduanya merupakan perangkat lunak sumber terbuka (open source software). Wordpress termasuk di dalam bagian blog. Blog merupakan singkatan dari web log adalah bentuk aplikasi web yang menyerupai tulisan-tulisan (yang dimuat sebagai posting) pada sebuah halaman web umum.[13]

2.10 Putty

Putty adalah sebuah program open source yang dapat digunakan untuk melakukan protokol jaringan SSH (Secure Shell), Telnet dan Rlogin. Protokol ini dapat digunakan untuk menjalankan sesi remote pada sebuah komputer melalui sebuah jaringan, baik itu LAN (Local Area Network), maupun internet. Program ini banyak digunakan oleh para pengguna komputer tingkat menengah atas, yang biasanya digunakan untuk menyambungkan, mensimulasikan atau mencoba berbagai hal yang terkait dengan jaringan.

Program ini juga dapat digunakan sebagai tunnel di suatu jaringan. Putty adalah software remote console/terminal yang digunakan untuk remote komputer dengan terhubungnya menggunakan port SSH atau sebagainya.[14]

2.11 Stickiness

Application Application load balancer merutekan setiap permintaan secara independen ke target terdaftar berdasarkan algoritma load-balancing yang dipilih. Namun, dapat menggunakan fitur sesi stickiness (juga dikenal sebagai afinitas sesi) untuk mengaktifkan penyeimbang beban untuk mengikat sesi pengguna ke target tertentu. Ini memastikan bahwa semua permintaan dari pengguna selama sesi dikirim ke target yang sama. Fitur ini berguna untuk server yang memelihara informasi status untuk memberikan pengalaman berkelanjutan kepada klien.

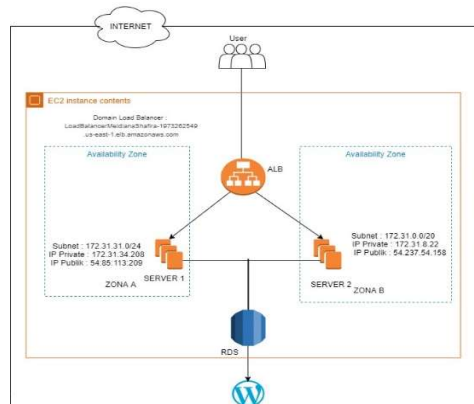
Application Application load balancer mendukung cookie berbasis durasi dan cookie berbasis aplikasi. Kunci untuk mengelola sesi stickiness adalah menentukan berapa lama penyeimbang beban harus secara konsisten mengarahkan permintaan pengguna ke target yang sama. Sesi lekat diaktifkan di tingkat grup target. Stickiness terdapat dua type kombinasi stickiness berbasis durasi, stickiness berbasis aplikasi.[15].

3. HASIL DAN PEMBAHASAN

3.1. Analisa Kebutuhan

Analisis sistem dilakukan untuk mengidentifikasi semua kebutuhan yang diperlukan untuk merancang dan membuat application load balancing menggunakan algoritma round robin dengan stickiness yang diterapkan pada amazon web services. Analisis dan perancangan application load balancer yang diimplementasikan pada sistem operasi Linux 2 dilakukan pada server yang dapat diakses dengan IP public yang disediakan oleh AWS dan untuk mengakses application load balancer diakses dengan domain application load balancer yang disediakan oleh AWS

a. Rancangan Topologi



GAMBAR 1. Tapologi Jaringan

Untuk merancang dan membangun application load balancer menggunakan AWS diperlukan sebuah rancangan topologi jaringan, berikut adalah topologi jaringan yang digunakan seperti pada gambar 1. Adapun Spesifikasi Hardware dan software yang digunakan antara lain :

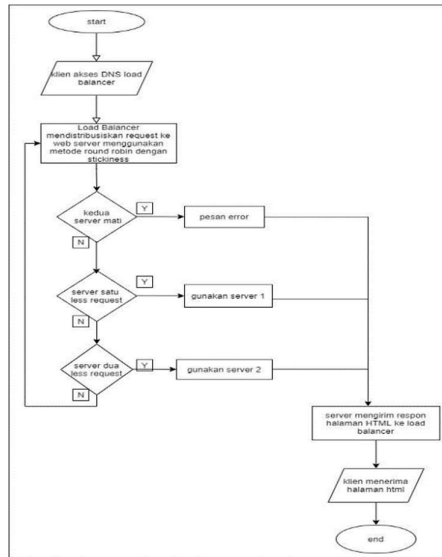
1. Server 1
 Instance type : T2.medium, Processor : Intel Xeon 2.4 GHz, Number of vCPUs : 2
 OS : Amazon Linux 2
 Web Server : Apache Web Server versi 2.4.48
 Lainnya : Wordpress versi 5.8
2. Server 2
 Instance type : T2.medium, Processor : Intel Xeon 2.4 GHz, Number of vCPUs : 2
 OS : Amazon Linux 2

Web Server : Apache Web Server versi 2.4.48
 Lainnya : Wordpress versi 5.8

3. RDS Database

DB Instance Class : Db.t2.micro, RAM : 1GB, Number of vCPUs : 1
 OS : Amazon Linux 2
 Engine Type : MySQL, Engine Version : MySQL 8.0.23

b. Diagram Sistem



GAMBAR 2. Alur Kerja Sistem

Diagram Sistem dari keseluruhan yang akan dibuat guna untuk memudahkan dalam proses pengerjaan project menggunakan algoritma round robin dengan stickiness pada aws. Pertama klien mengakses link website dari load balancer lalu load balancing mendistribusikan request ke web server menggunakan metode round robin dengan stickiness. Lalu load balancing akan mengecek kedua server, jika kedua server mati maka akan muncul pesan error Service Temporarily Unavailable. Jika server satu sedikit request maka load balancing akan mengarahkan ke halaman html server satu yang ada pada website blog wordpress, tetapi jika server dua yang memiliki sedikit request maka load balancing akan mengarahkan ke halaman html server dua yang ada pada website blog wordpress

Pada diagram kerja sistem diatas client akan mengakses domain load balancer, setelah itu load balancer akan menjalankan pembagian beban dengan algoritma round robin dengan stickiness, hasil load balancer akan terlihat pada server yang telah diinstallkan pada wordpress

3.2. Implementasi

Dalam pembuatan Application Load Balancing, membutuhkan dua buah server sehingga beban jaringan antara kedua server dapat terbagi rata. Langkah awal menyiapkan Instance di AWS serta konfigurasi. Berikut konfigurasi yang dilakukan pada instance AWS :

Konfigurasi Instance

```

#!/bin/bash sudo -i
yum update -y
sudo yum install -y httpd
sudo amazon-linux-extras install -y php7.2 yum install wget -y
wget http://repo.mysql.com/mysql-community-release-el6-5.noarch.rpm rpm -ivh mysql-community-release-el6-5.noarch.rpm
yum install mysql-server -y yum install php-mysqldb -y cd /var/www/html cho "healthy" > healthy.html
wget https://wordpress.org/latest.tar.gz tar -xzf latest.tar.gz
cp -r wordpress/* /var/www/html/
rm -rf wordpress
rm -rf latest.tar.gz chmod -R 755 wp-content
chown -R apache:apache wp-content chkconfig httpd on
service httpd start
    
```

Setiap instance memiliki ip public yang berbeda-beda, ip public berfungsi agar bisa diakses oleh semua orang melalui internet. Untuk mencegah ip public yang berubah-ubah maka diperlukan elastic ip yang tidak akan berubah walaupun instance tersebut diberhentikan.

Amazon Relational Database (Amazon RDS) digunakan untuk menyiapkan, mengoperasikan, dan menskalakan database relasional cloud. Amazon RDS memberikan keleluasaan untuk fokus pada aplikasi sehingga dapat memberikan performa kecepatan, ketersediaan, keamanan dan kompatibilitas tinggi yang di perlukan

3.3. Hasil dan Pengujian

TABEL 1. Pengujian Load Balancer dengan 5 perangkat			
Percobaan	Perangkat	Server 1	Server 2
1	Laptop 1		V
	Laptop 2	V	
	Smartphone 3		V
	Smartphone 4	V	
	Smartphone 5		V
2	Laptop 1	V	
	Laptop 2		V
	Smartphone 3	V	
	Smartphone 4		V
	Smartphone 5	V	
3	Laptop 1		V
	Laptop 2	V	
	Smartphone 3		V
	Smartphone 4	V	
	Smartphone 5		V

Dilakukan uji coba menggunakan 5 perangkat dengan tiga kali percobaan. Laptop 1 pada percobaan pertama mendapat hasil load balancer server 2, sementara pada percobaan kedua mendapat hasil application load balancer server 1 dan pada percobaan ketiga mendapatkan hasil application load balancer server 2. Laptop 2 pada percobaan pertama mendapat hasil application load balancer server 1, sementara pada percobaan kedua mendapat hasil load balancer server 2 dan pada percobaan ketiga mendapatkan hasil application load balancer server 1. Smartphone 3 pada percobaan pertama mendapat hasil application load balancer server 2, sementara pada percobaan kedua mendapat hasil application load balancer server 1 dan pada percobaan ketiga mendapatkan hasil application load balancer server 2. Dari hasil pengujian didapatkan hasil server 1 sebanyak 7 kali, dan server 2 muncul sebanyak 8 kali. Dari hasil pengujian tersebut application load balancer menjalankan pembagian beban kerja secara bergilir dari server 1 ke server 2 begitu juga sebaliknya.

TABEL 2. Pengujian Load Balancer dan Stickiness

No	Nama Pengujian	Skenario Pengujian	Hasil yang diharapkan	Hasil yang didapat	Hasil uji
1	Pengujian load balancer pada server dengan akses link domain	Mengakses link domain dan melihat hasil web server yang aktif pada website wordpress	Load Balancer akan mengarahkan ke server 1 atau ke server 2 berdasarkan jumlah user yang aktif	Load balancer berhasil mengarahkan user pada server 1 atau server 2 secara bergantian sesuai banyaknya user yang sedang aktif	Berhasil
2	Pengujian stickness dengan batas waktu	Pengujian Stickness dengan waktu 1 detik Pengujian Stickness dengan waktu 1 menit	Load balancer akan melakukan pembagian beban secara bergilir dari server 2 ke server 1 sesuai waktu yang di setting pada Stickness	Load balancer akan tetap diarahkan pada server tertentu selama masih dalam range waktu stickness. Jika rangenya sudah berakhir maka akan pindah ke server yang lain	Berhasil
3	Pengujian load balancer dengan menghentikan salah satu server dan tanpa mengklik menu website	Menghentikan Server 1 untuk mengecek load balancer mengarahkan user ke server 2	Load balancer akan mengarahkan user pada server yang aktif atau berjalan ketika salah satu server dihentikan jika belum ada mengklik menu di server	Load balancer berhasil mengarahkan user pada server yang aktif atau tetap bisa berjalan ketika salah satu server dihentikan jika belum ada mengklik menu di server	Berhasil
4	Pengujian load balancer dengan menghentikan salah satu server dan mengklik menu website	menghentikan salah satu server dan mengecek load balancer masih tetap berjalan atau tidak setelah mengklik salah satu menu pada website	Load balancer akan mengarahkan link website ke server yang aktif dan akan berada seterusnya di server tersebut, jika server tsb dimatikan maka load balancer tidak berjalan ke server yang lain jika sudah mengklik menu	Load balancer tidak berjalan seandainya sudah diklik salah satu menu di website, karena user tetap berada server yang sama / tidak akan pindah ke server yang lain, kecuali di mulai dari awal lagi	Berhasil

5	Pengujian load balancing dengan menghentikan kedua server	Menghentikan kedua server untuk mengecek load balancer masih berjalan atau error	Load balancer akan memunculkan pesan error “503 Service Temporarily Unavailable”	Load balancer berhasil memunculkan pesan error “503 Service Temporarily Unavailable”	Berhasil
---	---	--	--	--	----------

4. KESIMPULAN

Berdasarkan uraian permasalahan dan langkah kerja yang dilakukan dalam perancangan dan pembuatan application load balancing menggunakan algoritma round robin dengan stickiness pada aws, dapat diambil kesimpulan sebagai berikut :

1. Load balancing sangat berguna untuk pengelolaan server karena mampu membagi traffic antar server sama rata dan menghindari pembebanan berlebih pada satu server.
2. Penggunaan stickiness pada implementasi ini bertujuan untuk mengarahkan user pada target yang sama sesuai dengan waktu stickiness yang diterapkan, jika waktu stickiness masih berlangsung maka user akan tetap diarahkan pada target yang sama saat pertama akses application load balancer sementara jika waktu stickiness sudah habis maka application load balancer akan melakukan pembagian beban secara bergilir dari server 1 ke server 2 begitujuga sebaliknya.
3. Pengujian application load balancer saat salah satu server dihentikan bertujuan untuk melihat bahwa application load balancer akan mengarahkan user pada server yang aktif sehingga user masih tetap bisa mengakses web server tersebut.

REFERENSI

- [1] C. Kopparapu, Load balancing Servers , Firewalls , and Caches. .
- [2] F. Apriliansyah, I. Fitri, and A. Iskandar, “Implementasi Load balancing Pada Web Server Menggunakan Nginx,” *J. Teknol. dan Manaj. Inform.*, vol. 6, no. 1, 2020, doi: 10.26905/jtmi.v6i1.3792.
- [3] M. Santika and S. Hansun, “Implementasi Algoritma Shortest Job First dan Round robin pada Sistem Penjadwalan Pengiriman Barang,” *J. Ultim.*, vol. 6, no. 2, pp. 94–99, 2014, doi: 10.31937/ti.v6i2.336.
- [4] N. Sari, H. Amnur, and R. Hidayat, “Monitoring Next Cloud sebagai Private Cloud Storage dengan Notifikasi Telegram,” vol. 1, no. 4, pp. 144–149, 2020.
- [5] D. Lukitasari, F. Oklilas, F. I. Komputer, and U. Sriwijaya, “Analisis Perbandingan Load balancing Web Server Tunggal Dengan Web server Cluster Menggunakan Linux Virtual Server,” vol. 5, no. 2, pp. 31–34, 2010.
- [6] M. Ridwan and Y. Yustikasari, “Implementasi Teknologi Load Balancer Dengan Web Server Nginx Menggunakan Metode Ip Hash,” no. December, 2019.
- [7] A. Nugroho, W. Yahya, and K. Amron, “Analisis Perbandingan Performa Algoritma Round robin dan Least Connection untuk Load balancing pada Software Defined Network,” vol. 1, no. 12, pp. 1568–1577, 2017.
- [8] I. Susilo and G. Kristiyanto Nugraha, “Pembangunan Web Server Menggunakan Debian Server Untuk Media Pembelajaran Di Sekolah Menengah Kejuruan (Smk) Negeri 1 Sragen,” *Indones. J. Netw. Secur. (IJNS)-ijns.org IJNS*, vol. 2, no. 1, pp. 23–24, 2013, [Online]. Available: <http://kuis.smkn1srg.sch.id>.
- [9] T. Fajrin and S. M. K. N. Karanganyar, “Analisis Sistem Penyimpanan Data Menggunakan Sistem Cloud computing Studi Kasus SMK N 2 Karanganyar,” vol. 1, no. November, pp. 31–35, 2012.
- [10] A. W. Services, “Amazon Elastic Compute Cloud User Guide for Linux API Version 2015-04-15 Amazon Elastic Compute Cloud User Guide for Linux Amazon Elastic Compute Cloud : User Guide for Linux,” 2015.
- [11] Amazon Web Services, “Amazon Relational Database Service (RDS) User Guide,” *Amaz. Relational Database Serv. User Guid.*, 2020, [Online]. Available: <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Welcome.html>.

- [12] A. Aziz and T. Tampati, "Analisis Web Server untuk Pengembangan Hosting Server Institusi: Perbandingan Kinerja Web Server Apache dengan Nginx," *Multinetics*, vol. 1, no. 2, p. 12, 2015, doi: 10.32722/vol1.no2.2015.pp12-20.
- [13] N. Kurnia, D. Koryati, and R. Ar, "Pengaruh Media Wordpress Blog Terhadap Hasil Belajar Peserta Didik Pada Mata Pelajaran Ekonomi Di Sma Negeri 6 Palembang," *J. Profit*, vol. 2, no. 1, p. 48, 2015.
- [14] Z. Amin and D. Meldi, "Pengidentifikasian dan Pencarian Manusia Berbasis Citra Menggunakan Unmanned Aerial Vehicle," *Met. J. Sist. Mek. dan Termal*, vol. 2, no. 2, p. 50, 2018, doi: 10.25077/metal.2.2.50-60.2018.
- [15] N. L. Balancers, "Elastic Load balancing Network Load Balancers."
- [16] M. P. Index, "程威特 1 , 吴海涛 1 , 江帆 2," vol. 1, no. 10, 2021.
- [17] S. Halawa, "Perancangan Aplikasi Pembelajaran Topologi Jaringan Komputer Untuk Sekolah Menengah Kejuruan (SMK) Teknik Komputer Dan Jaringan (TKJ) Dengan Metode Computer Based Instruction," *JURIKOM (Jurnal Ris. Komputer)*, vol. 3, no. 1, pp. 66–71, 2016, [Online]. Available: <https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom/article/view/53/32>.