



Implementasi Cloud Computing Cluster Menggunakan Ovirt Untuk Layanan Web

Muhammad Arif Fadhly Ridha^{*1}, Thesa Nabila Balqis²

¹Teknik Informatika, Politeknik Caltex Riau, Pekanbaru, Indonesia

²Teknologi Rekayasa Komputer, Politeknik Caltex Riau, Pekanbaru, Indonesia

¹fadhly@pcr.ac.id, ²thesa22trk@mahasiswa.pcr.ac.id

**Corresponding Author*

Diserahkan: 19 Juni 2023

Diterima: 11 Juli 2023

Diterbitkan: 13 Juli 2023

ABSTRAK

Cloud Computing Cluster terdiri dari server yang berbeda dan bekerja sama untuk memastikan bahwa downtime dari sumber daya kritis dikurangi seminimal mungkin. Tujuannya yaitu memastikan bahwa sumber daya kritis mencapai ketersediaan maksimum yang disebut sebagai high availability. Salah satu permasalahan pada penerapan metode cluster cloud computing, yaitu kebutuhan platform untuk meningkatkan ketersediaan layanan. Salah satu platform yang dapat digunakan adalah Ovirt. Ovirt adalah solusi virtualisasi berbasis open source yang dirancang untuk mengelola seluruh infrastruktur instansi. Ovirt menggunakan hypervisor KVM dan dibangun di atas beberapa proyek komunitas lainnya, termasuk libvirt, Gluster, PatternFly, dan Ansible. Secara teknis Ovirt dapat memiliki teknologi Clustering dan memiliki High Availability pada environment yang menggunakan hypervisor lebih dari satu. Kegunaannya adalah jika salah satu hypervisor bermasalah maka Virtual Machine (VM) yang ada didalamnya dapat berpindah ke hypervisor lain. Hasil dari penelitian ini adalah tercapainya high availability pada layanan web server, dimana layanan web masih dapat diakses saat ada dua node dimatikan dan saat salah satu server dimatikan. Penggunaan CPU dan memori meningkat 1% ketika ada beberapa jumlah permintaan http request saat beberapa node dimatikan dan juga saat salah satu server fisik dimatikan.

Kata kunci: *Cloud Computing, Cluster, Ovirt*

ABSTRACT

A cloud computing cluster consists of different servers working together to ensure that downtime of critical resources is reduced to a minimum with the aim of ensuring that critical resources reach maximum availability, which is known as high availability. One of the problems in implementing the cluster cloud computing method is the need for a platform to increase service availability. One platform that can be used is Ovirt. Ovirt is an open-source-based virtualization solution designed to manage the entire infrastructure of an agency. Ovirt uses the KVM hypervisor and is built on top of several other community projects, including libvirt, Gluster, PatternFly, and Ansible. Technically, Ovirt can have Clustering technology and high availability in environments that use more than one hypervisor. Its usefulness is that if one hypervisor has a problem, the Virtual Machine (VM) in it can move to another hypervisor. The result of this research is the achievement of high availability on web server services, where web services can still be accessed when two nodes are turned off and when one of the servers is turned off. CPU usage increases when there are multiple HTTP requests, when multiple nodes are shut

down, and also when one of the physical servers is shut down. The same thing happens to memory usage, which increases by 1% when multiple nodes are shut down and also when one of the physical servers is shut down.

Keywords: *Cloud Computing, Cluster, Ovirt*

1. PENDAHULUAN

Perpustakaan Seiring dengan perkembangan teknologi yang semakin pesat, setiap orang atau instansi pasti membutuhkan informasi dan komunikasi untuk berbagai kepentingan sehari-hari saat ini. Berbagai informasi yang ingin didapatkan tentunya membutuhkan akses Internet. Teknologi Internet memberi kemudahan dan mempercepat setiap orang untuk mengakses berbagai informasi yang dibutuhkan. Bahkan melalui akses Internet, setiap orang akan lebih mudah untuk berkomunikasi secara cepat dan tepat.

Pemanfaatan jaringan Internet juga telah memberikan akses kemudahan bagi para pengguna dalam pembuatan *website*, *blog*, sistem informasi dan lain sebagainya yang menggunakan *web server*. Pada umumnya *website* tersebut dipasang pada sebuah *server* fisik yang juga menyimpan *website* lain sehingga tidak ada bagian khusus untuk sebuah *website*. Satu *website* akan berbagi *resource* dengan *website* - *website* yang lain. Sehingga ketika pengguna mengakses beberapa situs sekaligus maka *resource*, *service* dan *hardware* yang tersedia otomatis akan dipakai secara bersamaan. *Resource* yang dimaksud adalah prosesor, *storage*, RAM, dan *hardware*.

Selain itu, di dalam instansi masalah yang sering dihadapi adalah kurangnya *high availability* sehingga sewaktu-waktu layanan *web server* mendadak *downtime*, perkembangan infrastruktur Teknologi Informasi pun menuntut setiap instansi tersebut untuk melakukan inovasi - inovasi terkini, jumlah perangkat jaringan yang terhubung pun tentunya juga semakin meningkat. Maka dari itu dibutuhkan suatu sistem yang dapat membagi *resource* yang ada agar tiap-tiap *website* memiliki bagian *resource* khusus, maupun semua layanan dapat di akses secara *real time*, dan kapan saja, juga menjamin ketersediaan layanan yang tinggi karena semua data ditampung ke dalam *server* data yang *high availability*.

Cloud Computing dapat digunakan sebagai penyelesaian untuk semua permasalahan tersebut *Cloud Computing* adalah sekumpulan komputer yang bekerja sama dengan sekumpulan komputer lain yang ditempatkan pada lokasi geografis yang berbeda. Namun seiring dengan berkembangnya zaman *Cloud Computing* yang dikembangkan dengan menerapkan *cluster* jauh lebih baik, objek yang ada di dalam *cluster* memiliki kemiripan karakteristik antar satu sama lainnya dan berbeda dengan *cluster* yang lain. Partisi tidak dilakukan secara manual melainkan dengan suatu algoritma *clustering*. Oleh karena itu, *clustering* sangat berguna dan dapat menemukan *group* atau kelompok yang tidak dikenal dalam data. Dengan mengimplementasi kan sistem *Cloud Computing Cluster* pembagian tersebut tentunya dapat dilakukan, *resource* yang tersedia dapat dibagi ke beberapa virtual mesin. Tiap-tiap virtual mesin akan memiliki sistem operasi, RAM, dan *disc* tersendiri.

Salah satu teknologi yang dapat digunakan adalah *Ovirt* yang dapat memberikan kemudahan bagi para pengguna dalam mengakses berbagai informasi yang dibutuhkan melalui pemanfaatan Internet. Untuk mengatasi kelemahan dalam mengakses beberapa halaman *web* yang dilakukan sekaligus, umumnya sebuah instansi akan membeli perangkat *server* yang baru. Namun hal tersebut tidaklah efisien karena akan membuat ruangan pada instansi tersebut dipenuhi dengan perangkat *server*. Oleh karena itu, dibangunlah *web server* pada *cloud computing cluster* dengan menggunakan *Ovirt*.

2. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Pada penelitian ini diambil beberapa penelitian sebelumnya yang dapat dijadikan acuan dan landasan

awal. Dalam penelitian terdahulu ini diambil empat penelitian terdahulu yang sebelumnya telah dilakukan pengujian. Pada topik penelitian pertama, telah dilakukan oleh Setiya Budi (Budi, 2013) seorang mahasiswa Teknik Informatika Fakultas Sains Dan Teknologi Universitas Islam Negeri Sunan Kalijaga Kalijaga Yogyakarta, dengan judul “Implementasi Cluster Pada Web Server Berbasis Cloud Computing”. Menurut penelitian yang telah dilakukan oleh Setiya Budi ia menyatakan bahwa Rancangan sistem Cluster pada web server berbasis Cloud Computing ini berhasil dilakukan dengan tahapan penelitian Perancangan jaringan Implementasi rancangan sistem dan pengujian sistem. Pada implementasinya sistem ini mampu meminimalisir masalah downtime dengan nilai 10 detik pada saat terjadi proses failover[1].

Sistem Cluster pada web server juga telah berhasil dibangun diatas platform virtual berbasis Proxmox yang didalamnya terpasang sistem operasi Debian Squeeze pada platform KVM. Hasil yang muncul cukup mumpuni untuk skala mesin virtual dalam hal pelayanan yang mendekati mesin server riil. Pada topik penelitian kedua dilakukan oleh dilakukan oleh tiga orang mahasiswa yaitu Aziz Setyawan H, Eri Riana, dan Mohammad Nurohman (H & Dkk, 2016) yang berasal dari Program Studi Teknik Komputer AMIK BSI Tegal, AMIK BSI Bekasi, dan STMIK Nusa Mandiri Jakarta, mereka menyatakan bahwa Cloud computing merupakan teknologi yang sedang berkembang sangat pesat pada saat ini sehingga ada beberapa penyedia aplikasi open source seperti aplikasi Nextcloud yang dapat memberikan kendali penuh terhadap pengguna. Memiliki cloud server pribadi atau private cloud computing merupakan hal yang menguntungkan dan bermanfaat bagi suatu organisasi maupun perusahaan[2].

Cloud server computing dapat diakses melalui smartphone ataupun desktop, sehingga memudahkan pengguna dalam mengakses data yang mereka miliki. Selain dapat menyimpan data seperti upload dan download files, cloud server juga memberikan kemudahan lain seperti sharing files, menulis atau merubah dokumen secara langsung (real time), chatting, memutar video, musik, menyimpan nomor kontak, mengirim e-mail, menandai kalender membuka photo, dan memberikan informasi berita dengan melakukan shouting atau broadcast message.

Pada topik penelitian ketiga dilakukan oleh [2] dengan judul “analisis dan pembangunan infrastruktur cloud computing”. Infrastruktur yang dibangunnya adalah server as a service (SaaS) yang ditujukan untuk pemenuhan kebutuhan kegiatan praktikum siswa khususnya mata pelajaran jaringan komputer siswa kelas XI. Hasil dari penelitian ini yaitu prototype IaaS yang dimana prototype IaaS tersebut memiliki kinerja yang lebih baik dibandingkan dengan sistem yang sedang berjalan. Peneliti menggunakan Ubuntu Enterprise Cloud (UEC) sebagai aplikasi untuk membangun IaaS[3].

Pada topik penelitian ke empat dilakukan oleh Putu Gede Surya Cipta Nugraha (Nugraha, 2017) yang berjudul “implementasi dan analisis kinerja private cloud computing di lingkungan GDLN universitas udayana” yang melakukan penelitian implementasi dan analisis kinerja private cloud computing menggunakan platform openstack [4].

2.2 Landasan Teori

2.2.1 Cloud Computing

Cloud Computing adalah sistem komputerisasi berbasis jaringan/*internet*, dimana suatu sumber daya, *software*, informasi dan aplikasi disediakan untuk digunakan oleh komputer lain yang membutuhkan. Model *billing* dari layanan ini umumnya mirip dengan modem layanan publik. Ketersediaan sesuai kebutuhan, mudah untuk di kontrol, dinamik dan skalabilitas yang hampir tanpa limit adalah beberapa atribut penting dari *cloud computing*[5].

2.2.2 Node

Sistem komputer terdiri dari beberapa *node*, paling sederhananya terdiri atas dua *node*. *Node* adalah computer yang mandiri dimana artinya mampu untuk memproses tugas komputasi tanpa *node* lain[6].

2.2.3 Cluster

Cluster terdiri dari set komputer terhubung dengan kerja sama sehingga banyak hal tetapi dilihat sistemnya tunggal. Tidak seperti komputer jaringan, komputer *cluster* komputer memiliki setiap *node* set untuk melakukan tugas yang sama, dikendalikan dan dijadwal oleh perangkat lunak. Komponen *cluster* biasanya terhubung satu dengan yang lain melalui jaringan Local Area Network (LAN) dengan setiap *node* komputer digunakan sebagai *server* berjalan sendiri dari sebuah sistem operasi[7].

2.2.4 Ovirt

Ovirt sebagai salah satu platform yang dapat digunakan untuk cloud computing, *ovirt* menyediakan virtualisasi layanan server yang dapat dikelola melalui dashboard berbasis web. *Ovirt* juga dapat memberikan kemudahan bagi para pengguna dalam mengakses berbagai informasi yang dibutuhkan melalui pemanfaatan Internet[8].

2.2.5 High Availability

High Availability adalah karakteristik sistem untuk melindungi atau memulihkan dari gangguan ringan dalam kerangka waktu singkat dengan cara yang sebagian besar otomatis[9].

2.2.6 Uptime and Downtime

Uptime merupakan lamanya waktu ketika *service* atau aplikasi dalam keadaan tersedia. Downtime merupakan lamanya waktu ketika aplikasi atau *service* tidak tersedia. Biasanya dihitung dari outage terjadi sampai waktu ketika *service* atau aplikasi tersedia kembali[10].

2.2.7 Web Server

Web server merupakan *server* di *internet* yang melayani koneksi transfer lewat protocol HTTP. *Web server* merupakan salah satu hal yang terpenting dari *server* di *internet* dibandingkan *server* lainnya seperti *ftp server*, *e-mail server*, ataupun *news server*[11].

2.2.8 Virtualisasi

Virtualisasi adalah suatu aplikasi perangkat lunak yang dapat mensimulasikan sumber daya perangkat keras[12].

2.2.9 JMeter

JMeter atau The Apache JMeter™ adalah aplikasi open source berbasis Java yang dapat dipergunakan untuk performance test. JMeter bisa digunakan untuk melakukan load/stress testing *Web Application*, *FTP Application* dan *Database server* test[13].

2.2.10 Grafana

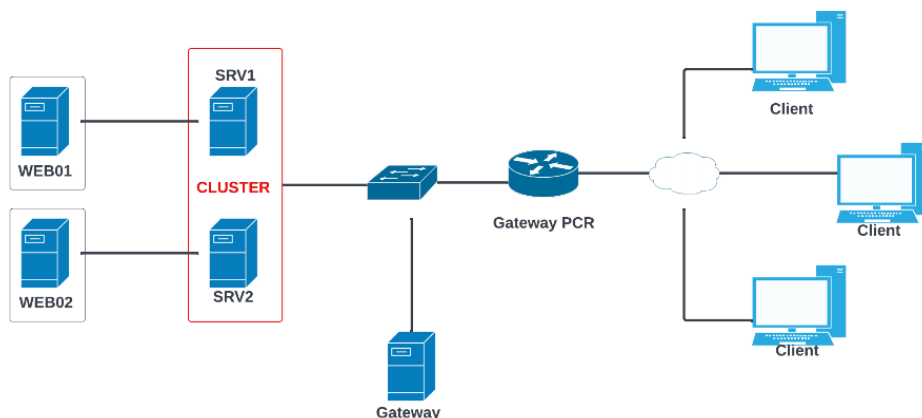
Grafana adalah sebuah software opensource yang membaca sebuah data metrics untuk dibuat menjadi sebuah grafik atau sebuah data tertulis. Grafana sering digunakan untuk melakukan analisis data dan monitoring[14].

3. METODE PENELITIAN

3.1 Topologi

Berikut adalah perancangan topologi jaringan yang akan di buat (gambar 1), beserta tabel pengalamatan yang akan diimplementasikan dalam proyek akhir ini.

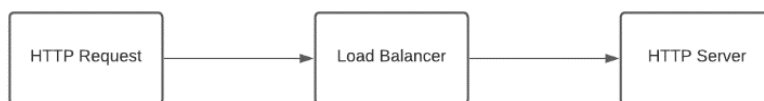
Rancangan topologi diatas ini menjelaskan, yaitu ada 2 *server* fisik terhubung ke jaringan publik milik Politeknik Caltex Riau dan setiap PC *Client* terhubung ke Internet sebagai pengujian terhadap *high availability* pada SRV1 dan SRV2. Pada topologi ini, SRV1 akan terhubung ke SRV2 dengan menggunakan teknik *Cluster* menggunakan *Ovirt*.



Gambar 1 Topologi

3.2 Architecture Logic

Berikut adalah architecture logic yang akan diimplementasikan dalam proyek akhir ini.



Gambar 2 Arsitektur Logis

3.3 Skenario Pengujian

Pada proyek akhir ini skenario yang terjadi adalah *client* dengan jumlah besar dapat mengakses layanan web yang tersedia dengan hasil yang diharapkan adalah terciptanya *High Availability*. Pertama beberapa *client* akan mengakses layanan web, lalu permintaan tersebut akan diteruskan ke *DNS server* yang berada diluar *cluster*. Setelah permintaan diterima oleh *DNS* permintaan akan diteruskan ke *server nginx* sebagai *balancer*, lalu *balancer* akan membagi rata permintaan tersebut dan diteruskan ke *web server* yang berada di dalam masing-masing *server* fisik. Pada kasus proyek akhir ini apabila salah satu *web server* atau server fisik mengalami *downtime* atau *error* maka layanan web masi tetap bisa berjalan dengan cara *client* akan dibagi rata ke *server* yang masi berfungsi.

3.4 Tabel IP

Berikut adalah Skenario IP yang akan digunakan dalam penelitian ini:

Tabel 1 Skenario IP

Perangkat	Interface	IP Address	Subnet Mask	Software
SRV1	Eth0	103.19.208.121	/25	Ovirt
SRV2	Eth0	103.19.208.22	/25	Ovirt
WEB01	Eth0	103.19.208.54	/25	Ubuntu + Apache2
WEB02	Eth0	103.19.208.55	/25	Ubuntu + Apache2
Gateway	Eth0	103.19.208.56	/25	Ubuntu + Nginx
PC-Client	Eth0	192.168.100.12	/24	Windows + Apache JMeter

3.5 Kebutuhan Perangkat

Proses penelitian pada proyek akhir ini membutuhkan beberapa perangkat dalam proses pengerjaan nya, baik perangkat keras maupun perangkat lunak sebagai berikut:

- 1) SRV1

Spesifikasi yang digunakan pada server dapat dilihat pada Tabel 2 dan Tabel 3 berikut.

Tabel 2 Spesifikasi SRV1

Spesifikasi	
Processor	8 Core
Hard Disk	100 GB
Memory	8 GB
Operating System	Ovirt 4.4
Software	Ovirt

2) SRV2

Tabel 3 Spesifikasi SRV2

Spesifikasi	
Processore	4 Core
Hard Disk	100 GB
Memory	8 GB
Operating System	Ovirt 4.4
Software	Ovirt

3) WEB1-2

Adapun spesifikasi web yang dibutuhkan pada penelitian ini dapat dilihat seperti Tabel 4 berikut.

Tabel 4 Spesifikasi Web Server

Spesifikasi	
Processore	1 Core
Hard Disk	20 GB
Memory	2 GB
Operating System	Ubuntu 20.04
Software	Apache2

4) Gateway

Adapun spesifikasi gateway yang diperlukan selama pengerjaan proyek akhir dapat dilihat seperti Tabel 5 berikut.

Tabel 5 Spesifikasi Gateway

Spesifikasi	
Processore	2 Core
Hard Disk	100 GB
Memory	8 GB
Operating System	Ubuntu 20.04
Software	NginX

5) PC-Client

Adapun spesifikasi PC-Client yang dibutuhkan pada penelitian ini dapat dilihat seperti Tabel 6 berikut.

Tabel 6 Spesifikasi PC-Client

Spesifikasi	
Processore	4 Core
Hard Disk	1 TB
Memory	10 GB
Operating System	Windows 10 Home
Software	Apache JMeter, Web Browser

3.6 Metode Pengujian

3.6.1 Pengujian High Availability

Pengujian ini dilakukan untuk melihat bagaimana kemampuan dari *web server* dalam menangani permintaan dari *client* saat seluruh *node* yang berada pada *server* fisik running dan saat hanya satu *node* yang running pada *server* fisik. Awalnya *client* mengakses *website* yang berada pada empat *node* dimana *node* tersebut berada pada masing-masing *server* fisik. Setelah itu, dua *node* dimatikan dan dilihat apakah *client* masih bisa mengakses *website* tersebut. Kemudian salah satu *server* fisik dimatikan dan dilihat apakah *client* masih bisa mengakses *website* tersebut.

3.6.2 Pengujian Performance

1) CPU

Pada penggunaan prosesor yang dilihat adalah persentase penggunaan prosesor saat *service* dijalankan dan diakses oleh *client*. Pada awalnya *web server* diakses oleh duapuluh *client* dan dilihat persentase penggunaan prosesor. Lalu, jumlah *client* ditambahkan sepuluh *client* menjadi tigapuluh *client* dengan tetap melihat persentase penggunaan prosesor. Setelah itu, tigapuluh *client* yang terhubung ditambahkan duapuluh *client* lagi menjadi limapuluh *client* dan tetap melihat persentase penggunaan prosesor. Hal yang sama juga akan dilakukan ketika satu *server* dimatikan. Pengujian ini akan dilakukan dua kali dalam sehari dan dilaksanakan selama satu minggu. *Tool* yang akan digunakan adalah JMeter dan Grafana.

2) RAM

Pada penggunaan RAM yang dilihat adalah penggunaan RAM saat web server dinyalakan. Pada hari pertama semua server akan dinyalakan sepanjang hari dan akan dilihat penggunaan memorinya. Pada hari ke dua dua web server dimatikan sementara yang lainnya tetap berjalan sepanjang hari dan akan dilihat penggunaan memorinya. Pada hari ketiga satu server fisik akan dimatikan dan yang lainnya akan tetap berjalan sepanjang hari. Semakin lama layanan web dinyalakan maka semakin naik penggunaan RAM nya. Pengujian ini akan dilakukan selama tiga hari. *Tool* yang akan digunakan adalah Grafana.

3) Pengujian web server Apache JMeter

Pengujian ini dilakukan untuk melihat layanan web server pada node– node yang ada pada server fisik. Apakah layanan web server dapat diakses oleh *client* yang berjumlah 100 *client* atau tidak. Pengujian pada web server juga dilakukan pada saat kondisi banyaknya pengguna yang mengakses ke web server sehingga dapat dilihat, apakah dengan menggunakan teknologi cloud computing tidak akan ada lagi masalah pada akses halaman web lain, dengan adanya pembagian yang merata ke dalam setiap halaman web memiliki masing masing web server dan memori yang sesuai dengan kebutuhan halaman web tersebut.

4. HASIL DAN PEMBAHASAN

4.1 Pengujian CPU

Berdasarkan pengujian terhadap *web server* pada masing-masing *server* menggunakan Jmeter, pengujian bertujuan untuk mengukur high availability yang dilakukan dengan 2 kali tahapan pengujian yaitu 100 *client* di pagi hari dan sore hari dengan masing-masing pengujian dilakukan 7 kali percobaan dalam mengambil data.

Tabel 7 Hasil pengujian CPU WEB01

WEB1	Pagi	Sore
Hari\User	100	100
1	0.7%	1.3%
2	0.7%	0.7%
3	0.7%	1.3%
4	0.7%	1.3%
5	1.3%	5.3%
6	0.7%	0.7%
7	0.7%	1.3%

Berdasarkan Tabel 7 diatas penggunaan CPU pada WEB01 mengalami peningkatan ketika jumlah permintaan diperbanyak, hal tersebut terjadi karena bertambahnya beban kerja pada web server untuk mengelola permintaan yang terus bertambah. Rata-rata yang didapatkan untuk pengujian ini adalah sebesar 0,78% untuk permintaan sebanyak 100 client di pagi hari dan 1,7% untuk permintaan di sore hari.

Tabel 8 Hasil pengujian CPU WEB02

WEB1	Pagi	Sore
Hari\User	100	100
1	1.3%	1.3%
2	1.3%	0.7%
3	4.7%	5.3%
4	0.7%	0.7%
5	0.7%	1.3%
6	0.7%	0.7%
7	0.7%	0.7%

Berdasarkan tabel diatas penggunaan CPU pada WEB02 mengalami peningkatan ketika jumlah permintaan diperbanyak, hal tersebut terjadi karena bertambahnya beban kerja pada web server untuk mengelola permintaan yang terus bertambah. Rata –rata yang didapatkan untuk pengujian ini adalah sebesar 1,44% untuk permintaan sebanyak 100 client di pagi hari dan 1,52% untuk permintaan di sore hari.

Berdasarkan table di atas dengan bertambahnya jumlah permintaan yang mengakses *web server*, maka penggunaan CPU pada *server* fisik dan *node* juga meningkat.

4.2 Pengujian RAM

- 1) Pada saat semua server dijalankan

Tabel 9 Hasil pengujian RAM hari pertama

DAY1	MEMORY USAGE
SRV1	46.50%
SRV2	18.00%
WEB01	13.00%
WEB02	13.00%
Gateway	13.00%
Hosted Engine	44.00%

Berdasarkan Pada tabel 9 merupakan kondisi pada data penggunaan memori yang normal jika semua *server* fisik dan *node* dijalankan. Persentase penggunaan memori pada setiap node terlihat masih minim

sekali. Dikarenakan pada hari pertama, sistem belum diuji oleh penggunaan yang banyak, sehingga sistem tidak terlalu banyak menggunakan resource.

- 2) Pada saat salah satu *web server* dimatikan

Tabel 10 Hasil pengujian RAM hari ketiga

DAY1	MEMORY USAGE
SRV1	56.00%
SRV2	16.00%
WEB01	no data
WEB02	16.00%
Gateway	15.00%
Hosted Engine	53.00%

Pada Tabel 10 merupakan data penggunaan memori ketika ada satu *node* yang tidak berjalan atau mati berdasarkan data tersebut dapat dilihat bahwa penggunaan memori pada SRV2 menurun sebanyak 2% dari penggunaan normal. Hal tersebut terjadi karena salah satu *node* yang terletak pada SRV2 tidak berjalan atau mati.

- 3) Pada saat SRV2 dimatikan

Tabel 11 Hasil pengujian RAM hari ketiga

DAY1	MEMORY USAGE
SRV1	57.00%
SRV2	N/A
WEB01	no data
WEB02	no data
Gateway	15.00%
Hosted Engine	54.00%

Pada Tabel 11 merupakan data penggunaan memori ketika SRV2 mati, berdasarkan data tersebut dapat dilihat bahwa penggunaan memori pada SRV1 meningkat sebanyak 1% dari penggunaan normal. Hal tersebut terjadi karena SRV2 tidak berjalan atau mati, sehingga SRV1 harus menjalankan *service* lebih untuk menjaga ketersediaan server. Dapat diperhatikan pula pada *node* WEB01 dan WEB02 akan ikut mati karena kedua *node* tersebut terletak pada SRV2.

5. KESIMPULAN

Dari hasil Pengujian dapat diambil kesimpulan sebagai berikut:

- 1) Layanan *web server* menggunakan *cluster cloud computing* dapat dibangun dengan Ovirt. Dengan menggunakan Ovirt, membuat beberapa server fisik atau virtual yang terhubung satu sama lain menjadi satu kelompok cluster. Kemudian, kelompok cluster ini dapat digunakan untuk menjalankan mesin virtual dan layanan web server yang dapat diakses oleh pengguna. Dengan menggunakan cluster cloud computing yang dibangun dengan Ovirt, layanan web server dapat diakses oleh banyak klien secara bersamaan tanpa terganggu oleh kegagalan pada satu atau beberapa node atau server fisik.
- 2) *High Availability* dapat tercapai, layanan *web server* tetap bisa di akses oleh client walaupun ada beberapa node dan server fisik yang mati
- 3) Persentase penggunaan CPU server fisik apabila salah satu server fisik didalam *cluster* dimatikan meningkat dari 56% sampai 57%, hal ini dapat terjadi karena meskipun server fisik yang dimatikan tidak lagi menjalankan tugasnya, beban kerja dan lalu lintas data tetap berjalan dan terdistribusi ke server fisik yang masih beroperasi. Dalam sebuah cluster, ketika satu atau beberapa server fisik dimatikan, beban kerja yang tadinya diperoleh oleh server yang dimatikan akan dibagi secara

merata ke server fisik yang masih beroperasi. Sehingga meskipun setiap server fisik dalam cluster diatur untuk bekerja pada batas yang sama, setelah dimatikan salah satu server fisik maka setiap server fisik yang tersisa dalam cluster menerima beban kerja yang lebih besar daripada sebelumnya.

- 4) Layanan web server dapat diakses oleh client yang berjumlah 100 *client*.

DAFTAR PUSTAKA

- [1] Budi, S. (2013). *Implementasi Cluster Pada Web Server Berbasis Cloud Computing*. Universitas Islam Negeri Sunan Kalijaga Yogyakarta
- [2] Ernawati, T. (2018). analisis dan pembangunan infrastruktur cloud computing. Politeknik TEDC Bandung.
- [3] H, A. S., & Dkk. (2016). *Cloud Server Computing*. AMIK BSI Tegal , AMIK BSI Bekasi , STMIK Nusa Mandiri Jakarta.
- [4] Nugraha, P. G. S. C. (2017). implementasi dan analisis kinerja private cloud computing di lingkungan GDLN universitas udayana. Universitas Udayana.
- [5] Sunyoto, Andi. 2018. Uji Kinerja Sistem Web Service Pembayaran Mahasiswa Menggunakan Apache JMeter (Studi Kasus : Universitas AMIKOM Yogyakarta). XIII: 44–52.
- [6] M. V. Sugianto, “Keuntungan Teknologi Virtualisasi & Cloud Computing,” 2011.
- [7] Baktikominfo, “BERKENALAN DENGAN NODE: PENGERTIAN, FUNGSI, VARIASI, DAN JENISNYA,” *BAKTIKOMINFO*, 2021.
- [8] S. van Vugt, *Pro Linux High Availability Clustering*. Berkeley, CA: Apress, 2014. doi: 10.1007/978-1-4842-0079-7.
- [9] oVirt, “Welcome to oVirt!,” 2013. <https://ovirt.org/community/about.html> (accessed May 01, 2023).
- [10] K. Schmidt, *High Availability and Disaster Recovery*, 1st ed. Frankfurt, Germany: Springer Berlin Heidelberg, 2006. doi: 10.1007/3-540-34582-5.
- [11] A. Suryanto, “Pembangunan Aplikasi Piket Pada Sekolah Menengah Atas Satu Kudus berbasis Web,” *Journal Speed – Sentra Penelitian Engineering dan Edukasi*, vol. 4, no. 4, pp. 29–33, 2012.
- [12] K. Hartanto, “Implementasi Virtual Private Server menggunakan Xen Hypervisor,” Universitas Diponegoro, 2012.
- [13] Apache Software Foundation, “Apache JMeter - Apache JMeter™,” *Apache JMeter™*, 2021. <https://jmeter.apache.org/>
- [14] Grafana, “Grafana: The open observability platform,” 2014. <https://grafana.com/> (accessed May 01, 2023).