

Analisis Performa Live Migration Pada Cloud Computing Dengan Metode Hybrid Menggunakan Openstack

1st Rissa Mellani Nur Wulandari

Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

rissamellan@student.telkomuniversity.ac.id

2nd Favian Dewanta

Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

favian@student.telkomuniversity.ac.id

3rd Arif Indra Irawan

Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

arifirawan@student.telkomuniversity.ac.id

Abstrak—Teknologi virtualisasi banyak digunakan di pusat data cloud dan infrastruktur TI saat ini. Salah satu dari teknologi kunci untuk virtualisasi server adalah live migrasi mesin virtual (VM). Live Migrasi merupakan perpindahan mesin virtual dari satu mesin worker node ke mesin worker node yang lain dengan keadaan terus menyala. Pada saat dilakukan dengan benar, maka prosesnya berlangsung tanpa efek yang nyata dari sudut pandang pengguna akhir. Teknologi ini memungkinkan VM untuk dipindahkan dari satu host fisik ke host lain sambil meminimalkan waktu henti layanan. hal tersebut merupakan permasalahan terkait topik tugas akhir ini. Dalam tugas akhir ini, yaitu dilakukan penelitian terkait menganalisis performa live migrasi pada VM di OpenStack dan menganalisis dampak dari tekanan dan kegagalan ini pada kinerja live migration VM. OpenStack merupakan seperangkat perangkat lunak untuk membangun dan mengelola platform komputasi awan untuk cloud publik dan pribadi serta merupakan kumpulan proyek perangkat lunak open source yang menyediakan solusi Infrastructure-as-a-Service (IaaS) melalui serangkaian layanan yang saling terkait. Pada tugas akhir ini dibahas mengenai bagaimana performa live migration menggunakan OpenStack antar 2 server yang dihubungkan dengan storage ceph, Diantara VM satu (controller-compute) dan Vm dua (compute) dibuat storage yang bisa membaca kedua node tersebut dari openstack dikarenakan instance openstack yang akan digunakan akan menggunakan volume dan volume instance tersebut tidak bisa menggunakan storage local, maka dari itu dibangun storage diantara vm satu dan vm 2 untuk bisa melakukan migrasi di antara virtual machine. lalu pada VM 1 di config beberapa component openstack seperti keystone, glance, nova, neutron dan horizon dikarenakan pada vm 1 dibuat sebagai controller-compute dan pada vm2 di config hanya component nova dan neutron dikarenakan pada vm 2 dibuat sebagai compute. setelah openstack dapat dibangun lalu pada instance openstack tersebut dibangun lah sebuah website untuk menguji fitur live migration pada openstack yang sudah dibangun lalu menguji kinerja infrastruktur tersebut berdasarkan parameter migration time, downtime dan ukuran data transfer pada mesin virtual dan menguji parameter-parameter pada Quality of Service (QoS). khususnya Delay, Throughput, Jitter dan packet loss. Lalu menganalisisnya. Untuk scenario pengujian pada live

migration yaitu terdapat 3 skenario yaitu pengujian live migration terhadap webserver pada video online, video offline dan tanpa video. Untuk hasil yang didapatkan dapat ditarik kesimpulan yaitu semakin besar ukuran data transfer maka semakin besar juga downtime dan migration time

Kata kunci— performansi jaringan, virtualisasi, live migration, openstack, Quality of Service (QoS), virtualisasi.

I. PENDAHULUAN

Seiring majunya perkembangan teknologi pada saat ini, terjadinya sebuah perkembangan pada teknologi dan informasi yang signifikan terutama pada kegiatan sehari-hari yang dapat dikerjakan dalam waktu singkat. Maka dari itu pengembangan teknologi berdasarkan internet pada saat ini lebih mengarah pada aplikasi system

Istilah *WebServices* sudah mulai berkembang dan mulai pada sudah banyak pula yang menggunakan. Dikarenakan kelebihan dari *web service* itu sendiri, diantaranya yaitu kemudahannya dalam pengiriman data, karena secara umum *web service* menggunakan protokol TCP/IP dan HTTP sebagai sarana komunikasinya. Seperti diketahui protocol HTTP merupakan protocol yang umum dipakai di internet.

Pada teknologi web service yang akan digunakan pada tugas akhir ini yaitu openstack, dikarenakan openstack ini menyediakan platform komputasi awan open source di mana-mana untuk cloud publik maupun pribadi, juga merupakan perangkat lunak gratis. OpenStack ini juga memberikan fasilitas untuk menyebarkan mesin virtual (VM) dan instance lain yang menangani tugas yang berbeda untuk mengelola lingkungan cloud, untuk mengakses openstack ini pengguna harus terkoneksi dengan internet agar bisa memasuki dan menjalankan situs tersebut. Jadi pengguna hanya cukup menyediakan sebuah pc

dan perangkat jaringan internet agar bisa terkoneksi dengan server internet dan menyimpan data di komputer server dan tidak harus menyediakan hard disk yang berkapasitas besar pada komputer tersebut. Maka dari itu, platform layanan harus online setiap saat untuk memberikan service kepada semua client, sedangkan platform juga pasti akan mengalami shutdown dan restart untuk keperluan maintenance. Live migration adalah suatu solusi yang dapat mengurangi downtime platform saat dilakukan maintenance dengan memindahkan service ke platform cadangan dalam keadaan menyala. Live migration mampu membekukan aplikasi yang sedang berjalan, melakukan record memory sesuai dengan keadaan aplikasi saat itu, memindahkan hasil record ke platform cadangan, restore data record dari platform utama di platform cadangan, dan melanjutkan aplikasi berjalan sesuai dengan keadaan pada saat checkpoint dibuat. Solusi ini sangat berguna untuk menjaga ketersediaan layanan pada saat platform harus berhenti sejenak

Metode live migration yang paling umum saat ini dapat dengan mudah dibedakan berdasarkan iterasi yang berbeda yaitu terdapat metode pre-copy (Memori ditransfer terlebih dahulu dan kemudian eksekusi ditransfer) dan metode post-copy (Eksekusi ditransfer terlebih dahulu dan kemudian memori). kedua metode tersebut memiliki kelemahan dan kekuatan sendiri. Metode pre copy tidak dapat berkinerja baik dalam lingkungan intensif tulis. Ini menunjukkan peningkatan drastis dalam total waktu migrasi. Di sisi lain, post-copy juga tidak berkinerja baik dengan read beban intensif. maka dari itu Metode live migration yang akan dilakukan yaitu metode hybrid. Metode ini akan mengambil yang terbaik dari masing-masing metode pre copy dan post-copy. Metode ini dapat menunjukkan bahwa pre-copy dan post-copy dapat berkinerja baik dalam skenario ini[1]

Dalam tugas akhir ini penulis menyelidiki keandalan live migration di openstack, dengan meningkatkan tekanan system selama migrasi. untuk permasalahan terkait topik tugas akhir ini, sebelumnya sudah pernah diangkat sebagai tugas akhir juga dengan judul yang berbeda yaitu dengan judul "LIVE MIGRATION PADA CLOUD COMPUTING DENGAN METODE HYBRID" pada tugas akhir tersebut dilakukan penelitian terhadap live migration pada cloud computing dengan menggunakan Kernel-based Virtual Machine (KVM) erbeda halnya dengan penelitian yang dilakukan pada buku ini yaitu live migration pada cloud computing dengan menggunakan Openstack. disini penulis akan mensimulasikan beberapa kegagalan jaringan seperti packet delay, jitter, Throughput. Yang dapat menentukan dampak kegagalan jaringan pada kinerja migrasi. Hasil dapat digunakan untuk memandu administrator pusat data dalam keputusan migrasi dan

lokalisasi kesalahan. Selain itu, dapat membantu untuk menemukan kemacetan dan metode pengoptimalan untuk live migration di openstack.

II. KAJIAN TEORI

A. Cloud Computing

Cloud computing sangat memiliki potensi untuk mengubah sebagian besar industri TI, Cloud Computing mengacu pada aplikasi yang dikirimkan sebagai layanan melalui Internet dan perangkat keras serta perangkat lunak systems di pusat data yang menyediakan layanan tersebut. Layanan itu sendiri telah lama disebut sebagai Perangkat Lunak sebagai Layanan (Software as a service). Beberapa vendor menggunakan istilah seperti IaaS (Infrastructure as a service) dan PaaS (Platform as a Service).

B. Web Service

Web Service adalah komponen perangkat lunak yang berdiri sendiri yang memiliki URI unik (*Uniform Resource Identifier* adalah alamat unik), dan yang beroperasi melalui Internet dan terutama Web. Premis dasarnya adalah bahwa *web service* memiliki penyedia dan memiliki pengguna atau pelanggan. Penyedia perlu menerapkan layanan, tetapi juga untuk menulis spesifikasi dan untuk membuat spesifikasi ini tersedia bagi calon pelanggan melalui publikasi dalam direktori.

C. Virtualisasi

Virtualisasi adalah platform teknologi dasar yang mendorong komputasi awan. Istilah virtualisasi mengacu pada abstraksi sumber daya komputasi (CPU, penyimpanan, jaringan, memori, tumpukan aplikasi, dan database) dari aplikasi dan pengguna akhir yang mengonsumsi layanan.

D. Live Migration

Live Migration adalah fitur yang sangat ampuh untuk cluster dan administrator cloud. Administrator dapat memigrasikan instans OS dengan aplikasi sehingga mesin dapat dibebaskan untuk Pemeliharaan. Demikian pula, untuk meningkatkan pengelolaan, instans OS dapat diatur ulang di seluruh mesin untuk meringankan beban pada kelebihan beban pada hosts dan dapat langsung melakukan live migration, cara kerja live migration tersebut yaitu pada saat status runtime harus ditransfer langsung dari sumber ke tujuan saat VM masih berjalan. Ada dua pendekatan utama yaitu *Post-Copy* dan *Pre-Copy* migrasi memori.

Parameter Live Migration

Terdapat 3 parameter live migration diantaranya total migration time, downtime, dan ukuran data transfer. Migration time adalah waktu dari awal proses migrasi langsung hingga kerangka kerja virtualisasi memberi tahu bahwa host sumber dapat dinonaktifkan. Downtime Ini didefinisikan sebagai waktu henti atau waktu blackout adalah fase selama migrasi ketika ada

ketidakterediaan layanan yang terlihat oleh pengguna. Total data yang ditransfer ini didefinisikan sebagai data yang ditransfer secara keseluruhan selama proses migrasi. Semakin banyak data ini berkurang, semakin banyak konsumsi jaringan berkurang

E. Openstack

OpenStack adalah seperangkat alat perangkat lunak untuk membangun dan mengelola platform komputasi awan untuk cloud publik dan pribadi serta merupakan kumpulan proyek perangkat lunak open source yang menyediakan solusi Infrastructure-as-a-Service (IaaS) melalui serangkaian layanan yang saling terkait.

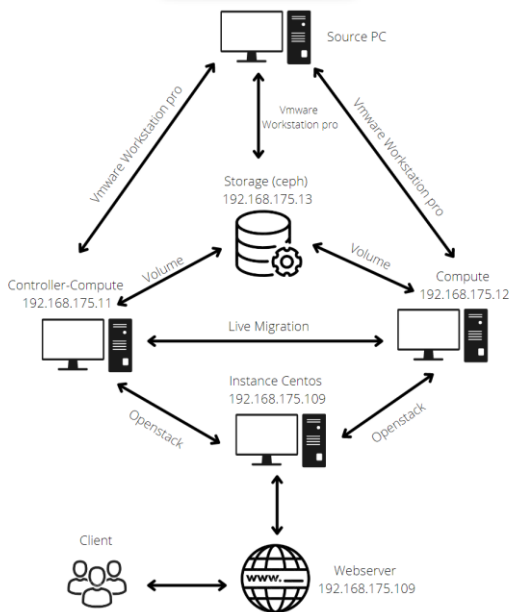
F. Ceph Storage

Open Source Ceph menyediakan beberapa layanan penyimpanan pada tingkat objek (Penyimpanan objek Ceph), tingkat blok (penyimpanan blok Ceph) dan tingkat file (sistem file Ceph).

III. METODE

A. Desain Sistem

Pada Tugas Akhir ini telah dilakukan perancangan *live migration* pada *cloud computing* menggunakan *openstack*.



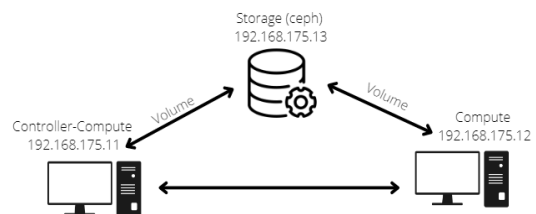
GAMBAR 3.1
DESAIN SISTEM

Cloud ini menggunakan 1 pc sebagai perangkat keras dengan tujuan membangun 3 virtual machine sebagai host vmsatu, (*controller-compute*), host vm 2 (*compute*) dan *storage (ceph)* lalu diantara kedua *virtual machine host* vm satu dan vm 2 tersebut dibentuk 1 instance yang dibentuk dari openstack dengan nama Centos-inst dan untuk mengontrol

Centos-inst ini digunakanlah openstack sebagai controller atau monitor virtual mesin. System operasi yang digunakan pada host dan target virtual machine adalah ubuntu 20.04.

Diantara VM satu (*controller-compute*) dan Vm dua (*compute*) dibuat storage yang bisa membaca kedua node tersebut dari openstack dikarenakan instance openstack yang akan digunakan akan menggunakan volume dan volume instance tersebut tidak bisa menggunakan storage local, maka dari itu dibangun storage diantara vm satu dan vm 2 untuk bisa melakukan migrasi di antara virtual machine. System operasi yang digunakan untuk instance yaitu image nya centos dan untuk system operasi pada vm 1, vm 2 dan storage adalah ubuntu 20.04. Perintah migrasi dapat dijalankan pada terminal vm 1 maupun vm 2, maka setelah proses migrasi selesai instance akan berpindah host sesuai yang dituju.

B. Perancangan Topologi Jaringan Cloud



GAMBAR 3.2
PERANCANGAN TOPOLOGI JARINGAN CLOUD COMPUTING

Beberapa komponen yang digunakan untuk mendukung perancangan sistem presensi pada penelitian Tugas Akhir ini adalah terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*).

C. Perancangan Mekanisme Migrasi

Pada Gambar3.3 merupakan flowchart yang menjelaskan alur algoritma hybrid dengan menjalankan mesin virtual, diantaranya

1. Persiapan migrasi dari host VM
2. Proses iterasi pre-copy dari sumber ke tujuan
3. Pemandahan CPU state dari sumber ke tujuan dan virtual machine mulai berjalan menuju host tujuan untuk menyelesaikan sisa memori ke host tujuan.
4. Proses migrasi selesai

D. Kebutuhan Hardware

Pada Tabel 3.1 terdapat spesifikasi laptop yang digunakan untuk membuat project tugas akhir ini dengan spesifikasi berikut

TABEL 3.1
SPESIFIKASI

Processor	11th Gen Intel(R) Core (i7-1165G7 @ 2.80GHz
Core	4 core 8 Threads

RAM	16 GB
Penyimpanan	512 GB SSD
Sistem Operasi	Windows 11
Hypervisor	VMware Workstation 16 Player

E. Parameter Live Migration

1. Downtime

Waktu yang diambil oleh proses migrasi untuk menghentikan virtual machine di sumber dan melanjutkan di host target. Ini secara langsung mempengaruhi ketersediaan layanan.

2. Data Transfer

Data Transfer adalah jumlah total halaman yang ditransfer selama migrasi. Untuk kinerja yang lebih baik, nilai ini harus lebih kecil dan idealnya, harus sama dengan jumlah total halaman virtual machine

3. Migration Time

Migration time adalah waktu yang dibutuhkan dalam menyelesaikan transfer virtual machine dari sumber ke target.

F. Parameter Quality of Service (QoS)

Quality of Service (QoS) dapat dikatakan sebagai suatu terminologi yang digunakan untuk mendefinisikan karakteristik suatu layanan (service) jaringan guna mengetahui seberapa baik kualitas dari layanan tersebut. Penyediaan QoS mencakup penyediaan Kualitas Layanan kepada pengguna akhir dalam hal beberapa parameter generik. Kualitas layanan yang dirasakan dapat diukur secara kuantitatif dalam hal beberapa parameter. Dalam analisis ini terdapat throughput, Average Delay, Average jitter dan packet loss yang dipertimbangkan.

ITU-T G.1010 merupakan standarisasi untuk layanan web yang dibuat oleh International Telecommunication Union untuk End-user multimedia QoS categories Standarisasi terhadap parameter tersebut dapat dilihat pada tabel dibawah

TABEL 3. 2
QOS

Parameter	Data Transfer
Throughput	10 KB/s - 10 MB/s
Delay	Preferred <2 s /page Acceptable <4 s /page
Packet Loss	Zero
Jitter	N.A.

1. Throughput

Throughput merupakan kemampuan jaringan mengirim data dengan menghitung rata rata pengiriman yang berhasil di dalam jaringan.

2. Delay

Delay merupakan waktu tunda dalam suatu pemrosesan data, maka dari itu delay ini akan menjadi waktu yang dibutuhkan oleh paket untuk melintang dari sumber ke tujuan.

3. Jitter

Jitter adalah perbedaan selang waktu kedatangan antar paket di terminal tujuan, atau dengan kata lain jitter merupakan variasi dari delay.

IV. HASIL DAN PEMBAHASAN

A. Kondisi Pengujian

Pada tahap ini terdapat beberapa *tools* yang digunakan dalam pembangunan *cloud*, antara lain *VMware Workstation 15.5 PRO*. Untuk mengimplementasikan penelitian ini dibutuhkan perangkat keras berupa komputer yang digunakan sebagai sumber untuk membuat *virtual machine* pada perancangan sistem *cloud*. Di dalam *VMware Workstation* akan dibuat 3 buah virtualmachine, yaitu Controller dan VM Dua. Kedua *virtual machine* ini menggunakan operasi sistem ubuntu server 20.04. Spesifikasi dari VM Satu, VM Dua dan VM tiga dapat dilihat pada tabel di bawah ini

TABEL 4. 1
SPESIFIKASI VM

Komponen			
	VM satu	VM dua	VM tiga
Hypervisor	Vmware Workstation Pro	Vmware Workstation Pro	Vmware Workstation Pro
Name	Controller-Compute	Compute	Storage
Hard Disk Size	20 GB	20 GB	160 GB
CPU	4	4	4
Memory	7 GB	4 GB	3 GB
Ip Address	192.168.175.11	192.168.175.12	192.168.175.13

Pada table diatas sebagai *hypervisor* dan menggunakan tambahan *opensource* yang bernama *Openstack* sebagai antarmuka untuk mengelola *virtual machine* yang akan dibuat pada VM Satu. Instance yang digunakan hanya satu, yaitu instance dengan menggunakan *image centos 7* Spesifikasi dari instance dapat dilihat pada tabel di bawah ini

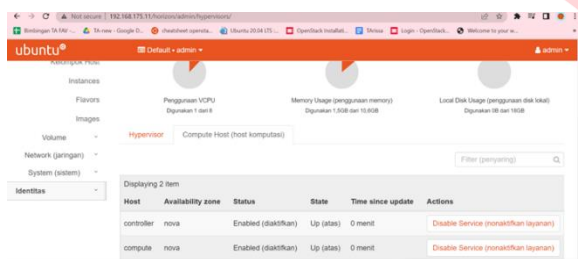
TABEL 4. 2
SPESIFIKASI INSTANCE

Name	Centos-ints
Flavor	512 MB
Image	Centos7

Volume	Vol-centos 7 11 GB
Security Group	Sec-Group
Keypair	Key1
Ip Address	192.168.175.109

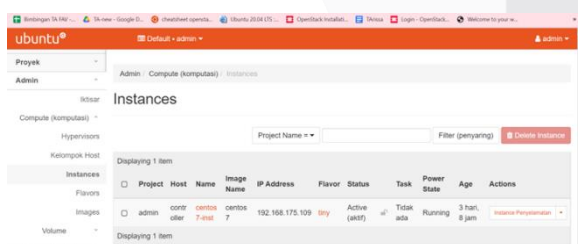
Cloud dibangun dengan menggunakan VMware Workstation untuk membuat virtual machine host dan virtual machine tujuan. Gambar tersebut menunjukkan sistem operasi virtual machine yang digunakan adalah ubuntu server 20.04. Selanjutnya melakukan penginstalan openstack per komponen pada masing masing VM, pada VM 1 dilakukan penginstalasi all in one controller dan compute, dan pada VM 2 dilakukan instalasi komponem compute dan untuk configuration controller nya menggunakan ip VM1.

Pada gambar dibawah sudah menunjukan bahwasanya sudah terdapat 2 compute yang dapat berkomunikasi sudah ter instal dengan baik setelah melakukan konfigurasi, untuk file file konfigurasi agal lebih jelasnya akan disertakan ada halaman



GAMBAR 4.1
COMPUTE

Pada gambar diatas sudah menunjukan bahwasanya sudah terdapat 2 compute yang dapat berkomunikasi sudah ter instal dengan baik setelah melakukan konfigurasi



GAMBAR 4.2
INSTANCE

Pada gambar diatas yaitu menunjukan bahwasanya instance sudah terbuat dan sudah berstatus running, memiliki Ip public 192.168.175.109. untuk instance tersebut diisikan oleh webserver yang pada akhirnya webserver tersebut akan dijadikan sebagai objek live migration. Pada saat melakukan live migration ini akan dilakukan penelitian pada parameter live migration terhadap 3 skenario pengambilan data. Untuk pengambilan data nya yaitu

terdapat pengujian live migration pada website dengan melakukan streaming video youtube secara online, pengujian live migration pada webserver dengan melakukan streaming pada video offline dan yang terakhir yaitu pengujian live migration pada website yang tidak disertai video.

Untuk pengujian live migration ini dilakukan pada website host controller-compute (VM1) menuju host compute (VM2) secara bulak balik dengan kondisi instance menyala (running)

B. Tujuan Pengujian

Sistem webservice yang telah dibangun melalui infrastructure of a service (IAAS) dengan menggunakan openstack dan telah dapat melakukan fitur live migration dengan metode hybrid pada Tugas Akhir ini telah diimplementasikan dan telah dirancang, maka langkah selanjutnya adalah dilakukan pengujian dan analisis terkait sistem penelitian. Tujuan dari pengujian ini adalah sebagai berikut:

1. Mengimplementasikan suatu system webservice yang memiliki kemampuan live migration.
2. Menguji kinerja *live migration* berdasarkan parameter *migration time*, ukuran *data transfer* dan downtime pada mesin virtual.
3. Mengukur dan melakukan pengujian performansi sistem yang dibuat yang berupa Delay, Throughput, Jitter, packet loss
4. memahami pengaruh dari live migration terhadap kualitas layanan yang sedang berjalan pada server virtual?

C. Skenario Pengujian

Untuk pengujian live migration ini dilakukan pada website host controller-compute (VM1) menuju host controller-compute (VM2) secara bulak balik dengan kondisi instance menyala (running).

1. Pengujian live migration pada website dengan 3 skenario

Pengujian dilakukan ketika instance diakses dan digunakan melalui laptop client dengan 3 skenario yaitu skenario pertama dengan menggunakan video streaming youtube yang telah diberikan pada website, skenario kedua yaitu dengan video offline yang sudah diunduh dan disimpan di website dan skenario yang ketiga yaitu tanpa video pada website Pengujian live migration ini dilakukan sebanyak 30 kali percobaan dari sumber host ke tujuan host tanpa merubah ip aslinya. Untuk menguji ketiga parameter live migration yaitu waktu migrasi, waktu downtime dan ukuran data transfer

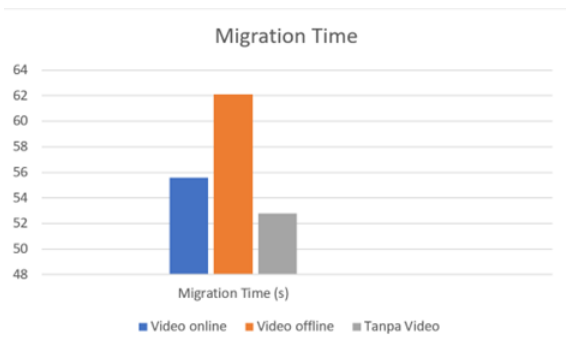
2. Pengujian Quality of Service (QOS)

Skenario pengujian QOS dilakukan dengan cara menggunakan wireshark, dan melakukan capture disaat waktu yang bersamaan dengan melakukan live migrasi setelah mengcapture 1515 packet data yang

sudah terfilter oleh ip website (192.168.175.109). parameter QOS yang akan digunakan yaitu Jitter, Delay, Packet Loss dan Throughput lalu hasil data tersebut akan dibandingkan dengan standarisasi QOS ITU-T G1010 untuk layanan Web.

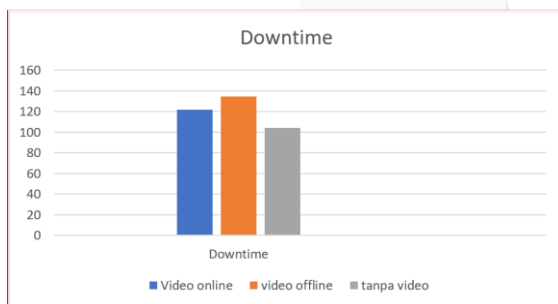
D. Hasil Pengujian

1. Pengujian terhadap parameter live migration Berikut merupakan grafik batang dari hasil pengujian live migration pada gambar 4.8 dengan scenario video online, video offline dan tanpa video pada website, pengujian ini dilakukan sebanyak 30 kali, dengan hasil nilai rata rata untuk waktu migrasi, downtime dan ukuran data transfer.



GAMBAR 4.3 MIGRATION TIME

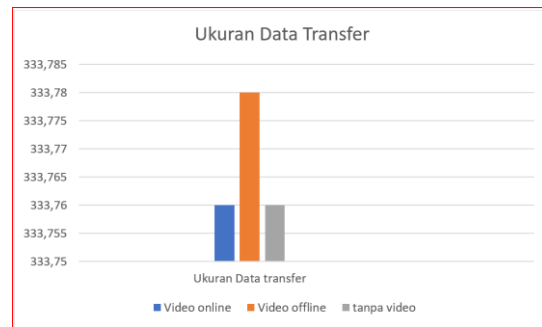
Gambar diatas merupakan grafik batang dari hasil pengujian *live migration* terhadap waktu migrasi. Terlihat pada scenario video offline memiliki waktu migrasi yang lebih Panjang dibandingkan video online dan tanpa video. Dan sebaliknya tanpa video merupakan waktu migrasi terpendek jika dibandingkan dengan 2 skenario tersebut.



GAMBAR 4.4 DOWNTIME

Grafik diatas merupakan grafik batang dari hasil pengujian *live migration* terhadap *downtime*, berdasarkan grafik tersebut sudah terlihat video offline

memiliki downtime yang tinggi dan tanpa video memiliki downtime terendah.



GAMBAR 4.5 UKURAN DATA TRANSFER

Pada gambar diatas menunjukkan grafik batang dari hasil pengujian *live migration* berdasarkan ukuran data transfer. Sudah terlihat video offline memiliki data transfer yang terbesar dibandingkan video online dan tanpa video.

1. Pengujian Quality Of Service (QoS)

Skenario pengujian delay dilakukan dengan cara menggunakan wireshark, dan melakukan capture disaat waktu yang bersamaan dengan melakukan *live migrasi* setelah mengcapture 1515 packet data yang sudah terfilter oleh ip website (192.168.175.109),

TABEL 4.3 HASIL QOS

Paramet er	Standarisasi ITU-T G1010	Hasil	Ketera- ngan
Through put	10 KB/s - 10 MB/s	0,4890 KB/s	Tidak Memenuhi
Delay	Preferred <2 s /page Acceptable <4 s /page	2 s	Memenuhi
Packet Loss	Zero	0,02%	Memenuhi
Jitter	N.A.	2 s	Memenuhi

Berdasarkan nilai yang tertera pada tabel diatas, untuk nilai throughput dengan 0,4890 KB/s belum memenuhi standar ITU-T G1010, dan untuk nilai delay mendapatkan hasil rata rata delay 2 s, packet loss 0,02% dan jitter 2s ketiga nilai tersebut telah memenuhi standar ITU-T G1010.

V. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat ditarik kesimpulan sebagai berikut:

1. Pengujian *live migration* ini dilakukan sebanyak 30 kali percobaan dengan 3 skenario yang berbeda. diantaranya yaitu untuk pertama yaitu *live*

migration pada webserver dengan menggunakan video online dan diperoleh dengan hasil rata rata waktu migrasi 55,6s, rata rata waktu downtime 121,8 s dan ukuran data transfer senilai 333,76 MiB. untuk pengujian yang kedua yaitu live migration pada webserver dengan menggunakan video offline dan diperoleh dengan hasil rata rata waktu migrasi 62,1 s, rata rata waktu downtime 134,5 s dan ukuran data transfer senilai 333,78 MiB, untuk pengujian yang ketiga yaitu live migration pada webserver tanpa video dan diperoleh dengan hasil rata rata waktu migrasi 52,8 s, rata rata waktu downtime 104,6 s dan ukuran data transfer senilai 333,76 MiB,

2. Hasil pengujian dengan menggunakan parameter live migration dapat berjalan pada proses live migration. Berdasarkan pengujian yang telah dilakukan sebanyak 30 kali, dengan hasil nilai rata rata untuk waktu migrasi, downtime dan ukuran data transfer. Pada skenario pengujian waktu migrasi, waktu downtime dan ukuran data transfer yaitu hasil yang diperoleh, video offline memiliki waktu migrasi, downtime dan ukuran data transfer yang lebih besar dibandingkan video online dan tanpa video. hasil tersebut dapat disimpulkan bahwasanya parameter waktu migrasi, downtime dan ukuran data transfer berbentuk garis lurus, semakin besar ukuran data transfer semakin besar pula waktu migrasinya, semakin besar waktu downtime akan semakin besar juga waktu migrasinya.
3. Dari skenario pengujian pada jaringan QOS, dapat disimpulkan dari hasil qos yang sudah didapatkan yaitu nilai throughput dengan 0,4890 KB/s maka hal tersebut belum memenuhi standar ITU-T G1010, dan untuk hasil yang didapat berdasarkan pengujian yaitu nilai delay mendapatkan hasil rata rata delay 2 s, packet loss 0,02% dan jitter 2 s ketiga nilai tersebut telah memenuhi standar ITU-T G1010 untuk layanan web.

REFERENSI

- [1] S. Sahni and V. Varma, "A hybrid approach to live migration of virtual machines," in 2012 IEEE international conference on cloud computing in emerging markets (CCEM). IEEE, 2012, pp. 1–5.
- [2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica et al., "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.
- [3] A. S. Rumale and D. Chaudhari, "Cloud computing: Infrastructure as a service," *International Journal of Inventive Engineering and Sciences*, vol. 1, no. 3, pp. 1–7, 2013.
- [4] S. Bhardwaj, L. Jain, and S. Jain, "Cloud computing: A study of infrastructure as a service (iaas)," *International Journal of engineering and information Technology*, vol. 2, no. 1, pp. 60–63, 2010.
- [5] G. Vossen and P. Westerkamp, "E-learning as a web service," in *Seventh International Database Engineering and Applications Symposium, 2003. Proceedings. IEEE, 2003*, pp. 242–249.
- [6] Y. Xing and Y. Zhan, "Virtualization and cloud computing," in *Future wireless networks and information systems*. Springer, 2012, pp. 305–312.
- [7] P. D. Patel, M. Karamta, M. Bhavsar, and M. Potdar, "Live virtual machine migration techniques in cloud computing: A survey," *International Journal of Computer Applications*, vol. 86, no. 16, 2014.
- [8] M. Noshay, A. Ibrahim, and H. A. Ali, "Optimization of live virtual machine migration in cloud computing: A survey and future directions," *Journal of Network and Computer Applications*, vol. 110, pp. 1–10, 2018.
- [9] F. Salfner, P. Troger, and A. Polze, "Downtime analysis of virtual machine live migration," in *The Fourth International Conference on Dependability (DEPEND 2011)*. IARIA, 2011, pp. 100–105.
- [10] R. Kumar, N. Gupta, S. Charu, K. Jain, and S. K. Jangir, "Open source solution for cloud computing platform using openstack," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 5, pp. 89–98, 2014.
- [11] V. Shamugam, I. Murray, J. Leong, and A. S. Sidhu, "Software defined networking challenges and future direction: A case study of implementing sdn features on openstack private cloud," in *IOP Conference Series: Materials Science and Engineering*, vol. 121, no. 1. IOP Publishing, 2016, p. 012003.
- [12] B. Cui and T. Xi, "Security analysis of openstack keystone," in *2015 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*. IEEE, 2015, pp. 283–288.
- [13] O. Tkachova, M. J. Salim, and A. R. Yahya, "An analysis of sdn-openstack integration," in *2015 Second International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T)*. IEEE, 2015, pp. 60–62.
- [14] V. Agrawal, D. Kotia, K. Moshirian, and M. Kim, "Log-based cloud monitoring system for openstack," in *2018 IEEE Fourth International Conference on Big Data Computing Service and Applications (BigDataService)*. IEEE,

- 2018, pp. 276–281.
- [15] D. Nguyen, J. Park, and R. Sandhu, “Adopting provenance-based access control in openstack cloud iaas,” in *International Conference on Network and System Security*. Springer, 2015, pp. 15–27.
- [16] R. Nasim and A. J. Kassler, “Deploying openstack: Virtual infrastructure or dedicated hardware,” in *2014 IEEE 38th International Computer Software and Applications Conference Workshops*. IEEE, 2014, pp. 84–89.
- [17] A. Sehgal, “Introduction to openstack,” *Running a Cloud Computing Infrastructure with OpenStack*, University of Luxembourg, 2012.
- [18] D.-Y. Lee, K. Jeong, S.-H. Han, J.-S. Kim, J.-Y. Hwang, and S. Cho, “Understanding write behaviors of storage backends in ceph object store,” in *Proceedings of the 2017 IEEE International Conference on Massive Storage Systems and Technology*, vol. 10, 2017.
- [19] E. Bugnion, S. Devine, M. Rosenblum, J. Sugerman, and E. Y. Wang, “Bringing virtualization to the x86 architecture with the original vmware workstation,” *ACM Transactions on Computer Systems (TOCS)*, vol. 30, no. 4, pp. 1–51, 2012.
- [20] M. Tabassum and K. Mathew, “Software evolution analysis of linux (ubuntu) os,” in *2014 International Conference on Computational Science and Technology (ICCST)*. IEEE, 2014, pp. 1–7.
- [21] W. Wood, “Migrating to mariadb.”
- [22] R. A. Talwalkar and M. Ilyas, “Analysis of quality of service (qos) in wimax networks,” in *2008 16th IEEE International Conference on Networks*. IEEE, 2008, pp. 1–8.