

**ANALISIS PERFORMA *NETWORK CLOUD*  
DENGAN METODE *QUALITY OF SERVICES*  
PADA PENYEDIA TEKNOLOGI LAYANAN *CLOUD COMPUTING* MICROSOFT  
AZURE DAN AMAZON WEB SERVICES**

***ANALYSIS OF NETWORK PERFORMANCE CLOUD  
WITH THE QUALITY OF SERVICES METHOD  
IN MICROSOFT AZURE AND AMAZON WEB SERVICES CLOUD COMPUTING  
TECHNOLOGY PROVIDER***

Abel Nathalia Widyastoro<sup>1</sup>, Rd Rohmat Saedudin, S.T., M.T<sup>2</sup>,  
Ahmad Almaarif, S.Kom, M.T<sup>3</sup>

<sup>1,2,3</sup>Prodi S1 Sistem Informasi, Fakultas Rekayasa Industri, Universitas Telkom

<sup>1</sup>abelwidyastoro@student.telkomuniversity.ac.id, <sup>2</sup>rdrohmat@telkomuniveristy.co.id,

<sup>3</sup>ahmadalmaarif@telkomuniversity.ac.id

---

**Abstrak**

*Beberapa tahun terakhir ini merupakan era layanan cloud dan mulai terlihat adanya pergeseran penggunaan layanan ke cloud. Penerapan cloud sendiri didasarkan pada penerapan teknik virtualisasi sehingga client tidak lagi menjadi sumber komputasi, melainkan dikerjakan pada suatu server yang berada pada jaringan internet (cloud). Teknologi cloud membutuhkan akses internet yang cepat dan dukungan transmisi data yang cepat pula. Hal ini menjadi tantangan baru yang diangkat oleh layanan cloud yaitu manajemen Quality of Service (QoS), yang harus menjamin bahwa setiap pengguna mendapatkan layanan aplikasi atau layanan cloud computing yang selalu tersedia dan sesuai dengan kebutuhan dari pelanggan. Tulisan ini akan berfokus pada pengujian performa network antara dua penyedia layanan cloud besar di dunia, yaitu Microsoft Azure dan Amazon Web Services. Metode yang digunakan dalam perbandingan tingkat performa adalah Quality of Services dengan parameter yang diuji adalah jitter, data loss, latency, dan throughput. Hasil yang didapat dari penelitian ini adalah perbandingan performa network dari masing-masing penyedia layanan cloud berdasarkan pengujian yang telah dilakukan sehingga dapat membantu pengguna yang akan memilih layanan cloud.*

**Kata Kunci :** *Quality of Service, cloud computing.*

---

**Abstract**

*The last few years have been the era of cloud services and are starting to appear shift in service usage to the cloud. The cloud application itself is based on the application of virtualization techniques so that the client is no longer a source of computing, but is done on a server that is on an internet network (cloud). Cloud technology requires fast internet access and support for fast data transmission. This is a new challenge raised by cloud services namely Quality of Service (QoS) management, which must guarantee that each users get application services or cloud computing services that are always on available and according to the needs of customers. This paper will focus on testing network performance between two major cloud service providers in the world, namely Microsoft Azure and Amazon Web Services. The method used in the comparison of performance levels is Quality of Services with the parameters tested are jitter, data loss, latency, and throughput. The results obtained from this study are the comparison of the network performance of each cloud service provider based on the testing that has been done so that it can help users who will choose cloud services.*

**Keyword :** *Quality of Service, cloud computing.*

---

**1. Pendahuluan**

Perkembangan teknologi *internet* yang pesat menghasilkan sistem komputasi baru yang disebut *cloud computing*. *Cloud computing* merupakan teknologi yang menghantarkan perangkat lunak dan juga sumber daya perangkat keras sebagai layanan melalui *internet* [1]. Dengan kata lain, *cloud computing* dapat meningkatkan ketersediaan dan fleksibilitas sebuah komputasi karena semua sumber daya komputasi yang kita miliki dapat dihantarkan kepada kita sebagai layanan di *internet*. Dalam beberapa tahun terakhir, *cloud*

*computing* memberikan dampak yang besar terhadap dunia informatika, contohnya perusahaan besar penyedia layanan seperti Amazon, Google, dan Microsoft. Perusahaan tersebut menawarkan layanan *cloud*, sehingga sebagian besar tugas pemantauan, konfigurasi, integrasi, pembelian infrastruktur perangkat keras dan perangkat lunak tidak lagi dibutuhkan karena akan ditangani oleh penyedia *cloud* [2].

Dengan berkembangnya dan tingginya permintaan untuk layanan komputasi ini serta dengan segala kemudahan dan kenyamanan yang ditawarkan, penyedia layanan *cloud computing* harus memperhatikan kualitas layanan yang diberikan. QoS menjadi sangat penting bagi pengguna *cloud*, yang mengharapkan penyedia untuk memberikan kualitas layanan yang sesuai dengan apa yang diiklankan, dan bagi penyedia untuk dapat menemukan nilai jual antara level QoS dengan biaya operasional yang dibutuhkan [3]. Maka dari itu, penting untuk mengukur QoS dari sebuah layanan baik dari pihak pengguna maupun pihak penyedia agar dapat tercapai tingkat kualitas layanan yang diinginkan.

Untuk meyakinkan pelanggan tentang performa *cloud* yang ditawarkan oleh penyedia layanan, perlu dilakukan *benchmarking* atau perbandingan performa terhadap masing-masing produk layanan *cloud* yang ditawarkan. Dari hasil *benchmarking*, kita dapat memperoleh gambaran mengenai produk mana yang paling sesuai dan efisien digunakan untuk pengguna.

Setelah meninjau beberapa poin yang sudah disebutkan sebelumnya, penelitian ini ditujukan untuk membantu pelanggan dalam memilih penyedia mana yang terbaik saat ini dari segi performa serta biaya yang ditawarkan. Melalui penelitian ini pun kita dapat melihat apakah layanan dan harga yang ditawarkan penyedia layanan sesuai dengan apa yang pelanggan dapatkan pada saat adopsi layanan *cloud* tersebut.

## 2. Dasar Teori

### 2.1 Definisi Cloud Computing

*Cloud Computing* adalah teknologi yang menghadirkan perangkat lunak komputasi serta sumber daya perangkat keras sebagai layanan melalui *internet* [1]. Komputasi *cloud* menyediakan akses ke *server*, penyimpanan, *database*, dan serangkaian layanan aplikasi yang luas melalui *internet*. Platform layanan *cloud* seperti Amazon Web Services memiliki dan melakukan pemeliharaan terhadap *hardware* yang terhubung dengan jaringan yang dibutuhkan untuk layanan aplikasi ini, sementara pengguna menggunakan apa yang dibutuhkan melalui aplikasi *web*.

#### 2.1.1 Karakteristik Cloud Computing

Berikut beberapa syarat layanan dapat dikatakan *Cloud Computing* [4]:

1. Layanan bersifat “*On Demand*”, pengguna dapat berlangganan hanya yang dibutuhkan saja, dan membayar hanya untuk yang digunakan saja.
2. Layanan bersifat elastis/*scalable*, di mana pengguna bisa menambah atau mengurangi jenis dan kapasitas layanan yang dia inginkan kapan saja dan sistem selalu bisa mengakomodasi perubahan tersebut.
3. Layanan sepenuhnya dikelola oleh penyedia/*provider*, yang dibutuhkan oleh pengguna hanyalah komputer personal/*notebook* ditambah koneksi *internet*.

#### 2.1.2 Klasifikasi Cloud Computing

*Cloud computing* dapat dikelompokkan menjadi tiga kategori menurut jenis *delivery service* sebagai berikut [5] :

##### A. Software as a Service

*Software as a Service* (SaaS) merupakan arsitektur *cloud computing* dimana jenis *service* ini berupa sejumlah aplikasi-aplikasi yang ditawarkan ke pihak pengguna. SaaS dapat diakses dari jarak jauh oleh pengguna melalui *internet* berdasarkan model harga yang ditawarkan. Layanan ini akan dikirimkan kepada para pengguna dengan menggunakan aplikasi penyedia yang berjalan dalam infrastruktur *cloud*. Aplikasi dapat diakses dari berbagai perangkat pengguna melalui *user interface* seperti *web browser*.

Seluruh infrastruktur *cloud* dikendalikan oleh pihak penyedia layanan termasuk jaringan, *server*, sistem operasi, dan media penyimpanan. Model ini dapat memberikan beberapa manfaat yang sangat menguntungkan baik bagi pengguna maupun penyedia jasa *cloud computing*.

##### B. Platform as a Service

Platform as a Service (PaaS) adalah layanan perangkat lunak perantara yang memfasilitasi program aplikasi-aplikasi lainnya di *environment cloud*. Kegunaan dari jenis *delivery service* ini yaitu untuk mendukung pengguna dalam pengerjaan aplikasi dengan menggunakan bahasa pemrograman dan *tools* tertentu yang didukung oleh penyedia layanan ke pengguna lainnya melalui fasilitas *cloud*. Para pengguna tidak mengelola atau mengendalikan infrastruktur secara fisik, termasuk jaringan, *server*, sistem operasi, ataupun media penyimpanan, namun mereka memiliki hak kendali secara penuh terhadap aplikasi yang sedang digunakan dan memungkinkan dilakukannya *hosting* konfigurasi aplikasi di lingkungannya.

Penyedia layanan PaaS menyediakan fasilitas untuk para *developer* untuk membuat aplikasi menggunakan bahasa pemrograman yang disediakan di *environment cloud* dengan seperangkat *Application Programming*

*Interface's* (API's) tertentu. PaaS akan memfasilitasi hubungan antar aplikasi *cloud*, serta mempercepat proses penyebaran dan mendukung skalabilitas yang diperlukan dari aplikasi *cloud* sendiri.

### C. *Infrastructure as a Service*

Infrastructure as a Service (IaaS) merupakan domain dari *hardware*, *software* dan perangkat jaringan lainnya. Kegunaan dari jenis *delivery service* ini adalah untuk meningkatkan kapasitas dalam hal pengelolaan sistem komputasi, seperti *server* dan unit pengolahan lainnya, peningkatan *storage*, peningkatan trafik jaringan, dan sumber daya lain dimana para pengguna dapat menyebarkan dan menjalankan *software* secara bebas, termasuk sistem operasi dan aplikasi. Pada dasarnya, para pengguna tidak mengetahui lokasi fisik perangkat yang mereka kelola. Namun, pengguna diberikan pengendalian penuh terhadap sistem komputasi, sistem operasi, media penyimpanan dan aplikasi yang mereka gunakan. Akan tetapi, pengguna dibatasi pengontrolannya yaitu hanya pada komponen jaringan yang dipilih.

## 2.2 *Quality of Services*

*Quality of Service* (QoS) merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari satu servis. QoS digunakan untuk mengukur sekumpulan atribut kinerja yang telah dispesifikasikan dan diasosiasikan dengan suatu servis[6].

### 2.2.1 Model Layanan *Quality of Services*

Terdapat tiga jenis model layanan *Quality of Service* (QoS), yaitu sebagai berikut [7]:

#### a. *Best-effort service*

*Best-effort service* adalah satu model layanan dimana aplikasi mengirim data setiap kali diharuskan dalam setiap kuantitas, dan tanpa meminta izin atau memberitahukan terlebih dahulu kepada jaringan. Untuk layanan *Best-effort service*, jaringan mengirimkan data jika bisa, tanpa jaminan kehandalan batas, atau *throughput*.

#### b. *Integrated service*

*Integrated service* adalah layanan beberapa model yang dapat menampung beberapa persyaratan QoS. Dalam model ini aplikasi meminta jenis layanan tertentu dari jaringan sebelum mengirim data. Aplikasi menginformasikan jaringan dari *traffic profile* dan meminta jenis layanan tertentu yang dapat mencakup *bandwidth* dan *delay requirement*. Aplikasi ini diharapkan untuk mengirim data hanya setelah mendapat konfirmasi dari jaringan.

#### c. *Differentiated service*

*Differentiated service* adalah layanan beberapa model yang dapat memenuhi persyaratan QoS yang berbeda. Namun, tidak seperti dalam model *integrated service*, aplikasi yang menggunakan *differentiated service* tidak secara eksplisit memberi isyarat *router* sebelum mengirim data.

### 2.2.2 Parameter QoS

QoS memiliki beberapa parameter yang dapat digunakan sebagai acuan untuk mengetahui kemampuan dari jaringan tersebut. Adapun parameter dari QoS yaitu [8]:

#### a. *Bandwidth*

*Bandwidth* merupakan lebar cakupan frekuensi yang digunakan oleh sinyal dalam medium transmisi. *Bandwidth* sering digunakan sebagai acuan untuk kecepatan pengiriman data.

#### b. *Throughput*

*Throughput* merupakan kemampuan sebenarnya suatu jaringan dalam melakukan pengiriman data. Umumnya, *throughput* dikaitkan dengan *bandwidth* dalam kondisi yang sebenarnya. *Bandwidth* lebih bersifat tetap sementara *throughput* sifatnya adalah dinamis tergantung lalu lintas data yang sedang terjadi. Beberapa faktor yang mempengaruhi *bandwidth* dan *throughput* antara lain perangkat jaringan, tipe data yang ditransfer, banyaknya pengguna jaringan, topologi jaringan, spesifikasi komputer *client/user*, spesifikasi *server* komputer, induksi listrik, cuaca dan lain-lain.

#### c. *Jitter*

*Jitter* adalah perubahan *latency* dari *delay* atau variasi waktu kedatangan paket. *Jitter* juga dapat diartikan sebagai gangguan pada komunikasi digital maupun analog yang disebabkan oleh perubahan sinyal karena referensi posisi waktu. Adanya *jitter* ini dapat mengakibatkan hilangnya data, terutama pada pengiriman data berkecepatan tinggi.

#### d. *Packet Loss*

*Packet loss* adalah parameter yang menunjukkan jumlah total paket yang hilang. Paket yang hilang ini dapat terjadi karena *collision* dan *congestion* pada jaringan. *Packet loss* dapat terjadi karena kesalahan yang diperkenalkan oleh medium transmisi fisik. Hal-hal yang dapat mempengaruhi *packet loss* antara lain kondisi geografis seperti kabut, hujan, gangguan radio frekuensi, sel *handoff* selama *roaming*, dan interferensi seperti pohon-pohon, bangunan, dan pegunungan.

e. *Latency*

*Latency* adalah total waktu tunda suatu paket yang diakibatkan oleh proses transmisi dari satu titik ke titik lain yang menjadi tujuannya. *Delay* di dalam jaringan terdiri dari *delay processing*, *delay packetization*, *delay serialization*, *delay jitter buffer* dan *delay network*.

### 3. Pengujian dan Analisis

#### 3.1. Pengujian

##### 1. Pengujian Azure

Pengujian performa *network* pada penyedia layanan Azure dilakukan menggunakan tiga size VM yang berbeda, yaitu B1ms, B2s, dan B2ms.

Berikut merupakan data yang didapat setelah melakukan uji coba pada *size* B1ms.

Tabel 1 Tabel hasil uji coba B1ms

B1ms				
Test	Jitter (ms)	Data Loss (percent)	Latency (usec)	Throughput (Mbps)
1	0,023	0,0043%	442,439	926,61
2	0,018	0%	434,645	926,22
3	0,004	0,0048%	445,684	926,90
4	0,013	0,0019%	449,542	926,52
5	0,012	0,00075%	263,551	926,65
Rata2	0,014	0,00235%	407,172	926,58

Berikut merupakan data yang didapat setelah melakukan uji coba pada *size* B2s.

Tabel 1 Tabel hasil uji coba B2s

B2s				
Test	Jitter (ms)	Data Loss (percent)	Latency (usec)	Throughput (Mbps)
1	0,006	0,00097%	461,966	995,82
2	0,004	0,00034%	461,647	995,04
3	0,016	0%	460,037	996,41
4	0,027	0,00110%	462,173	995,84
5	0,020	0,00022%	468,898	995,54
Rata2	0,0146	0,000526%	462,9442	995,73

Berikut merupakan data yang didapat setelah melakukan uji coba pada *size* B2ms.

Tabel 2 Tabel hasil uji coba B2ms

B2ms				
Test	Jitter (ms)	Data Loss (percent)	Latency (usec)	Throughput (Mbps)
1	0,064	0,0006%	324,450	996,34
2	0,024	0,0014%	549,194	995,40
3	0,022	0%	547,389	995,85
4	0,006	0,0017%	505,732	995,81
5	0,020	0,0012%	348,084	996,12
Rata2	0,0272	0,00098%	454,9698	995,904

## 2. Pengujian AWS

Pengujian performa *network* pada penyedia layanan AWS dilakukan menggunakan tiga *size* VM yang berbeda, yaitu *t2.small*, *t2.medium*, dan *t2.large*.

Berikut merupakan data yang didapat setelah melakukan uji coba pada *size* *t2.small*.

Tabel 3 Tabel hasil uji coba *t2.small*

t2.small				
Test	Jitter (ms)	Data Loss (percent)	Latency (usec)	Throughput (Mbps)
1	0,018	0,0053%	318,615	527,6
2	0,025	0,01%	321,78	144,23
3	0,034	0,00095%	321,178	140,32
4	0,032	0,0053%	326,577	159,68
5	0,023	0,012%	320,878	144,98
Rata2	0,0264	0,00671%	321,8056	223,362

Berikut merupakan data yang didapat setelah melakukan uji coba pada *size* *t2.medium*.

Tabel 4 Tabel hasil uji coba *t2.medium*

t2.medium				
Test	Jitter (ms)	Data Loss (percent)	Latency (usec)	Throughput (Mbps)
1	0,064	0,019%	345,424	597,56
2	0,066	0,0095%	334,827	278,18
3	0,053	0,00095%	337,998	280,78
4	0,107	0,0048%	354,152	301,39
5	0,032	0,00024%	357,134	330,97
Rata2	0,0644	0,006898%	345,907	357,776

Berikut merupakan data yang didapat setelah melakukan uji coba pada *size* *t2.large*.

Tabel 5 Tabel hasil uji coba *t2.large*

t2.large				
Test	Jitter (ms)	Data Loss (percent)	Latency (usec)	Throughput (Mbps)
1	0,017	0%	261,449	737,34
2	0,142	0%	261,037	667,14
3	0,033	0%	265,972	559,21
4	0,016	0%	275,39	561,55
5	0,019	0%	261,022	546,97
Rata2	0,0454	0%	264,974	614,442

### 3.2. Hasil Analisis

#### 1. Jitter

Berikut merupakan tabel hasil rata-rata perbandingan *jitter* antara Azure dan AWS.

Tabel 6 Tabel perbandingan jitter

Jitter (ms)					
Azure			AWS		
B1ms	B2s	B2ms	t2.small	t2.medium	t2.large
0,014	0,0146	0,0272	0,0264	0,0644	0,0454

Dari tabel perbandingan di atas, kedua penyedia layanan menghasilkan nilai *jitter* yang sangat kecil, yaitu hampir mendekati angka nol dan merupakan hasil yang sangat baik. Berikut merupakan perbandingan *jitter* dalam bentuk grafik.



Grafik 1 Grafik perbandingan jitter

Jika dilihat perbandingannya, terlihat bahwa semua *size* Azure memiliki nilai *jitter* lebih rendah dibandingkan dengan AWS. Untuk B1ms mengungguli t2.small, B2s mengungguli t2.medium, dan B2ms mengungguli t2.large. Dapat dilihat pula jika Azure memiliki nilai *jitter* yang lebih stabil dibandingkan dengan AWS karena perbedaan nilai yang cukup banyak dari setiap *size*.

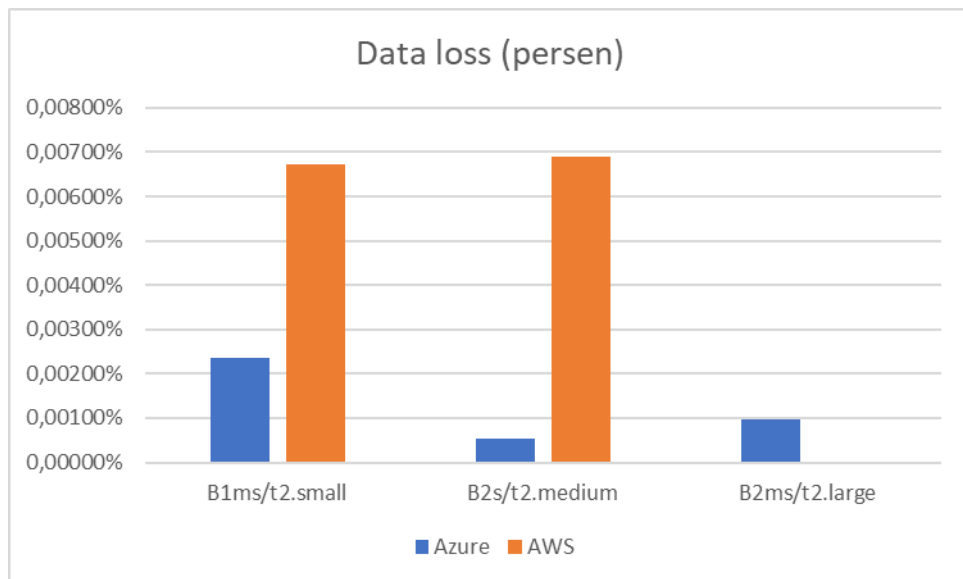
#### 2. Data loss

Berikut merupakan tabel hasil rata-rata perbandingan *data loss* antara Azure dan AWS.

Tabel 7 Tabel perbandingan data loss

Data Loss					
Azure			AWS		
B1ms	B2s	B2ms	t2.small	t2.medium	t2.large
0,00235%	0,000526%	0,00098%	0,00671%	0,006898%	0%

Dari tabel perbandingan di atas, kedua penyedia layanan menghasilkan nilai *data loss* yang sangat kecil, yaitu hampir mendekati angka nol dan merupakan hasil yang sangat baik. Berikut merupakan perbandingan nilainya dalam bentuk grafik.



Grafik 2 Grafik perbandingan data loss

Jika dilihat perbandingannya, nilai *data loss* t2.small lebih besar dari pada B1ms, t2.small jauh lebih besar dibandingkan dengan B2s, sementara B2ms lebih besar dibandingkan dengan t2.large. Hal ini menunjukkan bahwa *large size* AWS lebih baik dibanding Azure. Sementara untuk *size medium* dan *small*, Azure masih lebih baik dibanding AWS.

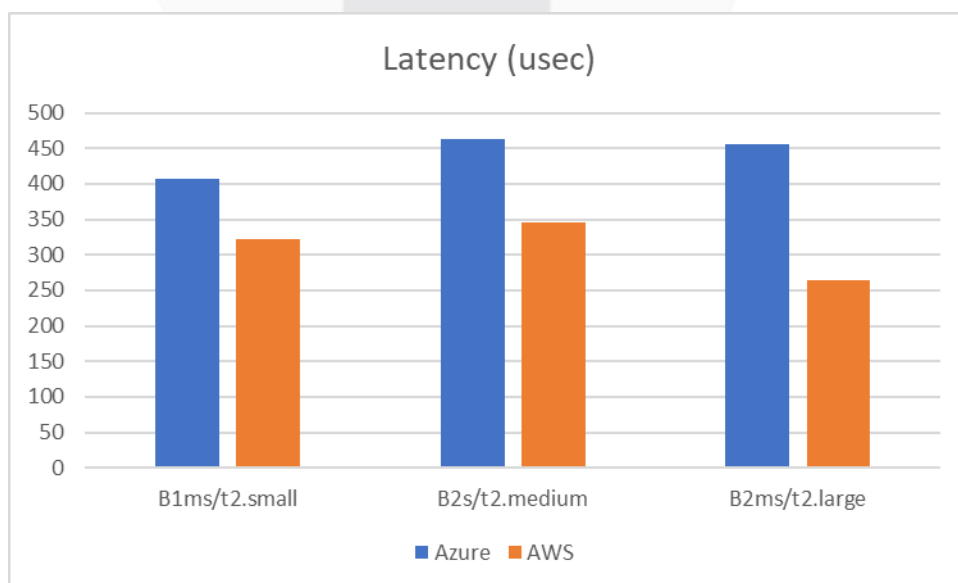
3. *Latency*

Berikut merupakan tabel hasil rata-rata perbandingan *latency* antara Azure dan AWS.

Tabel 8 Tabel perbandingan latency

Latency (usec)					
Azure			AWS		
B1ms	B2s	B2ms	t2.small	t2.medium	t2.large
407,1722	462,9442	454,9698	321,8056	345,907	264,974

Dari tabel perbandingan di atas, seluruh *size* Azure menghasilkan nilai *latency* yang hampir sama, dan untuk AWS terlihat menghasilkan nilai yang hampir sama untuk *size small* dan medium tetapi mengalami penurunan nilai *latency* saat menggunakan *size large*. Berikut merupakan data nilai *latency* dalam bentuk grafik.



Grafik 3 Grafik perbandingan latency

Jika dilihat perbandingannya, terlihat bahwa nilai *latency* B1ms lebih besar dibanding t2.small, lalu B2s juga memiliki nilai *latency* yang lebih besar dari t2.medium, sementara untuk B2ms nilainya jauh lebih besar dibanding t2.large. Hal ini menunjukkan bahwa semua *size* AWS memiliki nilai yang lebih baik dari Azure dengan nilai *latency* yang lebih kecil.

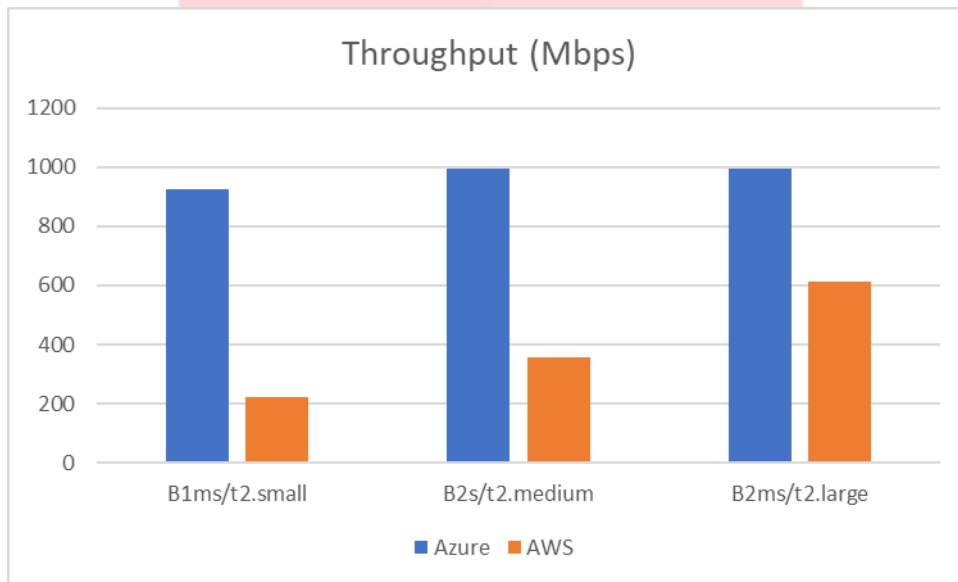
4. *Throughput*

Berikut merupakan tabel hasil rata-rata perbandingan *throughput* antara Azure dan AWS.

Tabel 9 Tabel perbandingan *throughput*

Throughput (Mbps)					
Azure			AWS		
B1ms	B2s	B2ms	t2.small	t2.medium	t2.large
926,58	995,73	995,904	223,362	357,776	614,442

Dari tabel perbandingan di atas, seluruh *size* Azure menghasilkan nilai *throughput* yang hampir sama, dan untuk AWS terlihat terdapat peningkatan *throughput* seiring dengan meningkatnya *size* yang digunakan. Berikut merupakan data nilai *throughput* dalam bentuk grafik.



Grafik 4 Grafik perbandingan *throughput*

Jika dilihat perbandingannya, terlihat bahwa nilai *throughput* dari B1ms jauh lebih besar dibanding t2.small, lalu nilai B2s juga jauh mengungguli t2.medium, dan untuk B2ms juga memiliki nilai *throughput* yang mengungguli t2.large. Hal ini menunjukkan bahwa seluruh *size* dari Azure mengungguli AWS dari segi *throughput* yang dihasilkan.

3.3. Analisis Hasil Keseluruhan

Setelah dilakukan perbandingan masing-masing parameter antara Azure dan AWS EC2, tabel berikut menampilkan penyedia mana yang unggul dari masing masing parameter yang dilakukan pengujian.

Tabel 10 Tabel hasil akhir perbandingan

Size	Jitter (ms)	Data Loss (percent)	Latency (usec)	Throughput (Mbps)
Small	Azure	Azure	AWS	Azure
medium	Azure	Azure	AWS	Azure
Large	Azure	AWS	AWS	Azure

Dari tabel diatas terlihat bahwa Azure mengungguli AWS di hampir semua parameter kecuali pada *latency* dan *data loss* pada *size large*. Data ini dapat menjadi pertimbangan bagi pengguna yang ingin menggunakan



*cloud computing* dengan fokus jaringan untuk memilih Azure karena performa *network* nya yang jauh mengungguli AWS.

Tabel 12 Tabel penilaian TIPHON

Penyedia	Size	Indeks Nilai				Kategori
		Jitter	Data loss	Latency	Throughput	
AWS	t2.small	3	4	4	4	Sangat bagus
	t2.medium	3	4	4	4	Sangat bagus
	t2.large	3	4	4	4	Sangat bagus
Azure	B1ms	3	4	4	4	Sangat bagus
	B2s	3	4	4	4	Sangat bagus
	B2ms	3	4	4	4	Sangat bagus

#### 4. Kesimpulan

Setelah dilakukannya pengujian masing-masing penyedia layanan dan memperoleh hasil analisis uji coba performa jaringan menggunakan metode QoS, dapat diambil kesimpulan sebagai berikut:

1. Dalam melakukan analisis performa jaringan cloud dengan metode analisis QoS dimulai dengan merancang *environment cloud* dengan tiga size yang berbeda pada setiap penyedia layanan untuk menguji masing-masing performa *size* nya. Untuk setiap *size* sendiri menggunakan masing-masing dua *virtual machine* dengan salah satunya akan berperan sebagai *server* dan yang lain menjadi *client*. Setelah itu melakukan uji coba dengan menggunakan empat parameter QoS untuk melihat performa jaringan *cloud*. Setiap parameter akan diuji dengan *tools* yang berbeda dengan total *tools* pengujian adalah tiga *tools*. Pada *tools* iPerf3 yang diperhatikan adalah *jitter* dan *data loss*, pada *sockperf* yang diperhatikan adalah *latency* dan pada *nttcp* yang diperhatikan adalah *throughput*. Sebelumnya masing-masing *virtual machine* harus ditempatkan di *Availability Zone* yang sama agar dapat terkoneksi satu sama lain. Pada tahap pengujian, dilakukan testing masing-masing parameter sebanyak lima kali dengan durasi testing 300 detik.
2. Setelah dilakukannya pengujian, dilakukan perbandingan untuk masing-masing *size* dari setiap penyedia layanan untuk melihat penyedia mana yang memiliki performa terbaik dari setiap parameter. Dari lima hasil testing yang dilakukan pada setiap parameter, diambil hasil rata-ratanya dan langsung dibandingkan hasil performanya. Hasil akhirnya dapat dilihat pada tabel V.10, dimana terlihat bahwa Azure mengungguli AWS pada hampir semua parameter kecuali parameter *latency* yang menandakan bahwa performa jaringan pada *environment cloud* Azure lebih baik dibandingkan dengan AWS.
3. Dari segi harga, Azure juga mengungguli AWS dengan *price/hour* yang lebih murah untuk *virtual machine* yang kemampuannya setara. Namun untuk pilihan *size* yang disediakan, AWS mengungguli Azure dengan pilihan *size* yang lebih banyak dan membuat AWS lebih fleksibel untuk segi harga tergantung dengan kebutuhan pelanggan.

#### Daftar Pustaka:

- [1] Sajjad, Muhammad. 2018. *Performance Evaluation of Cloud Computing Resource*
- [2] Ashari, Ahmad. 2011. *Cloud Computing : Solusi ICT ?*
- [3] Anderson, Helen. 2015. *A Survey on Quality of Service in Cloud Computing*.
- [4] Heryananda, Nichita. 2018. Mengenal Cloud Computing Lebih Jauh. Diakses melalui <https://diskominfo.kaltimprov.go.id/mengenal-cloud-computing-lebih-jauh/>
- [5] Afdhal. 2013. Studi Perbandingan Layanan Cloud Computing. Banda Aceh.
- [6] Wulandari, Rika. 2016. Analisis Qos (Quality Of Service) Pada Jaringan Internet (Studi Kasus : Upt Loka Uji Teknik Penambangan Jampang Kulon – Lipi). Sukabumi.
- [7] Suhervan. 2010. Analisis Penerapan QOS (Quality Of Service) pada jaringan fram Relay Menggunakan Cisco Router. Jakarta: Universitas Esa Unggul.
- [8] Sofana, Iwan. 2011. Teori dan Modul Praktikum Jaringan Komputer. Bandung: Modula.

