sudo
Jurnal Teknik Informatika
https://jurnal.ilmubersama.com/index.php/sudo

Computer Network

Analisis Kinerja Protokol Routing Open Shortest Path First (OSPF) pada Jaringan Universitas Islam Sumatera Utara

Kiki Finata *, *Khairuddin Nasution*

Fakultas Teknik, Program Studi Teknik Informatika, Universitas Islam Sumatera Utara, Medan, Indonesia

INFORMASI ARTIKEL

Diterima Redaksi: 03 Juni 2024
Revisi Akhir: 26 Juni 2024
Diterbitkan *Online*: 30 Juni 2024

KATA KUNCI

UISU; Topologi Jaringan; OSPF;
Routing Protokol; WAN

KORESPONDENSI

Phone: +62 895-3260-66199
E-mail: kiki_finata0899@gmail.com

A B S T R A K

Dalam lingkungan Universitas Islam Sumatera Utara (UISU), Jaringan *Wide Area Network* (WAN) memegang peran yang sangat penting dalam mendukung operasional dan kolaborasi antara fakultas serta unit administratif. Masih mengandalkan *routing* statis sebagai metode utama untuk mengelola *rute* komunikasi di jaringannya. Namun, untuk meningkatkan efisiensi dan keandalan jaringan, penelitian ini bertujuan untuk beralih ke Protokol *Routing* OSPF (*Open Shortest Path First)* sebagai metode *routing* dinamis yang lebih adaptif. Salah satu masalah utama adalah keterbatasan *fleksibilitas* dan *responsivitas* saat terjadi perubahan dalam topologi jaringan atau penambahan perangkat. OSPF memungkinkan pengenalan otomatis terhadap perubahan ini, mengurangi kebutuhan akan pembaruan manual yang rumit pada setiap *router*. Kemampuan OSPF dalam mengatasi kegagalan jaringan secara otomatis juga menjadi keunggulan penting. Dalam situasi di mana keandalan jaringan adalah suatu keharusan, OSPF dapat mendeteksi dan merespons kegagalan dengan cepat, meminimalkan waktu henti dan menjaga kelancaran layanan jaringan.

PENDAHULUAN

Dalam lingkungan Universitas Islam Sumatera Utara (UISU), Jaringan *Wide Area Network* (WAN) memegang peran yang sangat penting dalam mendukung operasional dan kolaborasi antara fakultas serta unit administratif. Saat ini, UISU masih mengandalkan *routing* statis sebagai metode utama untuk mengelola *rute* komunikasi di jaringannya. Namun, untuk meningkatkan efisiensi dan keandalan jaringan, penelitian ini bertujuan untuk beralih ke Protokol *Routing* OSPF (*Open Shortest Path First)* sebagai metode *routing* dinamis yang lebih adaptif.

Penggunaan OSPF diantisipasi akan memberikan solusi terhadap beberapa tantangan yang dihadapi dalam penggunaan *routing* statis saat ini. Salah satu masalah utama adalah keterbatasan *fleksibilitas* dan *responsivitas* saat terjadi perubahan dalam topologi jaringan atau penambahan perangkat. OSPF memungkinkan pengenalan otomatis terhadap perubahan ini, mengurangi kebutuhan akan pembaruan manual yang rumit pada setiap *router*. Menurut Nugroho et al., 2023, protokol *routing* OSPF lebih mudah diterapkan karena tidak perlu mendaftarkan atau mengkonfigurasikan setiap router ke semua *network* yang menghubungkan *network* asal ke tujuan. Penggunaan topologi WAN juga memiliki kelebihan tersendiri sebagaimana yang ditunjukkan salah satu kelebihan WAN yaitu dapat menghubungkan semua orang dengan menggunakan data lalu lintas yang sama.

Jaringan komputer adalah jaringan telekomunikasi yang memungkinkan antar komputer untuk saling berkomunikasi dengan bertukar data. Jaringan komputer terdiri dari perangkat lunak, perangkat keras dan media penghubung yang ketiganya saling terintegrasi satu sama lainnya. [1].

https://doi.org/10.56211/sudo.v3i2.532

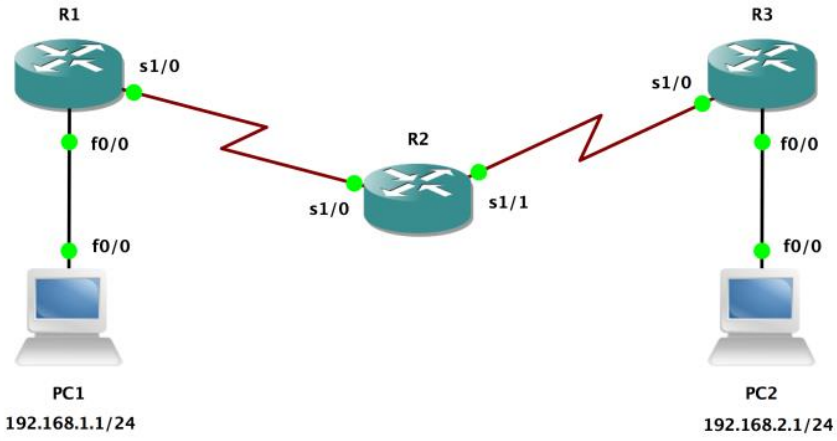
OSPF (Open Shortest Path First) adalah sebuah protokol routing dinamis yang digunakan dalam jaringan komputer untuk menemukan jalur terpendek dan paling efisien antara router-router dalam suatu jaringan. OSPF termasuk dalam kategori protokol link-state, yang berarti setiap router dalam jaringan OSPF akan membangun peta topologi jaringan secara lengkap dan menggunakannya untuk menentukan rute terbaik. [2].

Menurut Artha et al (2023) topologi terbagi atas beberapa jenis yaitu Topologi *Point-to-Poin*t menghubungkan dua perangkat secara langsung, membentuk koneksi khusus antara keduanya. Sementara itu, topologi Bus menghubungkan semua perangkat ke saluran komunikasi bersama, memungkinkan data yang dikirim oleh satu perangkat dapat mencapai semua perangkat lain. Walaupun efisien secara biaya, topologi *bus* rentan terhadap kegagalan jaringan karena satu kerusakan pada saluran komunikasi dapat mempengaruhi seluruh jaringan [3].

METODOLOGI

Topologi Jaringan

Topologi merupakan suatu tatanan untuk menghubungkan beberapa komputer atau perangkat-perangkat jaringan yang digunakan menjadi suatu jaringan yang saling terhubung. Jaringan komputer dapat diatur dalam berbagai topologi, masing-masing dengan keunggulan dan kelemahan uniknya, topologi terbagi atas beberapa jenis yaitu Topologi *Point-to-Poin*t menghubungkan dua perangkat secara langsung, [4].



Gambar 1. Rancangan Topologi

Pengaturan OSPF

Pengaturan OSPF melibatkan konfigurasi protokol OSPF pada perangkat jaringan yang terlibat dalam jaringan *Wide Area Network* (WAN). Berikut ini adalah beberapa pengaturan umum yang perlu dipertimbangkan saat mengkonfigurasi OSPF yakni Jaringan OSPF biasanya dibagi menjadi beberapa area untuk mempermudah manajemen dan skalabilitas. Konfigurasi OSPF menggunakan area *backbone* atau area 0 dengan wajib mendaftarkan area router yang berada di bawahnya berdasarkan alamat sesuai main router dan backup router [5],[6].

Sebuah protokol routing dinamis yang digunakan dalam jaringan komputer untuk menemukan jalur terpendek dan paling efisien antara *router-router* dalam suatu jaringan. OSPF termasuk dalam kategori protokol *link-state*, yang berarti setiap *router* dalam jaringan OSPF akan membangun peta topologi jaringan secara lengkap dan menggunakannya untuk menentukan rute terbaik [7]-[8].

Beberapa karakteristik utama OSPF adalah:

1. *Link-State* Protocol: OSPF menggunakan algoritma *link-state* untuk membangun peta topologi jaringan dan menentukan jalur terpendek ke setiap tujuan.
2. *Hierarchical Design*: OSPF memungkinkan pembagian jaringan menjadi area-area yang lebih kecil untuk mengurangi *overhead routing* dan meningkatkan efisiensi.
3. *Fast Convergence*: OSPF cepat dalam mendeteksi perubahan topologi jaringan dan memperbarui *rute*, sehingga jaringan dapat beradaptasi dengan cepat terhadap perubahan.

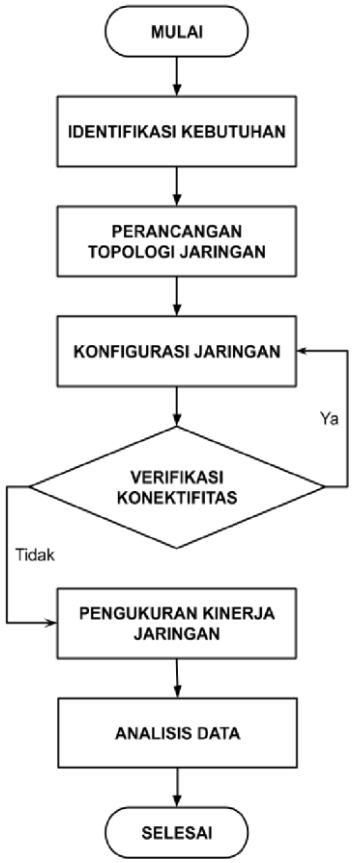
4. *Use of Cost Metric*: OSPF menggunakan *cost* (biaya) sebagai metrik untuk menentukan jalur terpendek. Cost biasanya dihitung berdasarkan bandwidth link.
5. *Support for CIDR and VLSM*: OSPF mendukung penggunaan *Classless Inter-Domain Routing* (CIDR) dan *Variable Length Subnet Mask* (VLSM), yang memungkinkan alokasi alamat IP lebih efisien.
6. *Authentication*: OSPF mendukung mekanisme otentikasi untuk memastikan keamanan pertukaran informasi routing antara *router*.

Dalam OSPF, setiap *router* akan mengirimkan LSA (*Link-State Advertisement*) yang berisi informasi tentang status dan biaya dari *link-link* yang terhubung ke router tersebut. *Router* kemudian menggunakan informasi ini untuk membangun peta topologi jaringan dan menjalankan algoritma Dijkstra untuk menemukan jalur terpendek ke setiap tujuan dalam jaringan. OSPF banyak digunakan dalam jaringan perusahaan dan penyedia layanan internet (ISP) karena skalabilitasnya, efisiensi, dan kemampuan untuk beradaptasi dengan berbagai topologi jaringan.

Kerangka Kerja Penelitian

Dalam penelitian ini, pendekatan eksperimental digunakan untuk menguji kinerja protokol *routing* OSPF pada jaringan *Wide Area Network* di UISU. Desain penelitian terdiri dari beberapa tahap yang terstruktur. Pertama, dilakukan identifikasi kebutuhan dan rumusan masalah untuk memahami tujuan dan fokus penelitian. Selanjutnya, topologi jaringan WAN yang sesuai dipilih berdasarkan kebutuhan penelitian, dan rancangan topologi jaringan yang memadai disusun.

Setelah itu, dilakukan konfigurasi pada *router* dan *switch* dalam jaringan sesuai dengan rancangan yang telah ditetapkan. Tahap berikutnya adalah verifikasi konektivitas, di mana pengujian dilakukan untuk memastikan bahwa semua perangkat dalam jaringan dapat saling berkomunikasi dengan benar. Setelah konektivitas terverifikasi, dilakukan pengukuran kinerja jaringan menggunakan alat bantu seperti *Graphical Network Simulator* 3. Pengukuran dilakukan untuk mengamati parameter seperti *delay*, *throughput* dan *packet loss*. Data hasil pengukuran dianalisis dan dievaluasi untuk mendapatkan pemahaman yang mendalam tentang kinerja protokol routing OSPF pada jaringan WAN yang terdapat di USIU. Untuk lebih mudah memahami tahapan yang ada berikut gambar *flowchart* yang digunakan pada penelitian ini.



Gambar 2. Flowchat Penelitian

 https://doi.org/10.56211/sudo.v3i2.532

HASIL DAN PEMBAHASAN

Alamat pada jaringan yang dirancang menggunakan IPv4 yang disesuaikan dengan kebutuhan jaringan itu sendiri. Untuk masing - masing hubungan antar router digunakan ip awal kelas a dimana masing-masing *router* memiliki 2 antarmuka untuk saling terhubung. Agar dapat memberikan efisiensi pada perencanaan alokasi alamat maka digunakan panjang prefix /30 dengan penjelasan sebagai berikut.

Tabel 1. Data Topologi

Alamat IP Awal	: 10.10.10.0/24 (Kelas A)
Jumlah Router	: 6
Jumlah subnet	: 6 router * 2 antarmuka/router = 12 subnet
Panjang Prefix Subnet Mask	: (/30) → 2^{32-30} = 4 - 2 = 2 (Jumlah Host) : (/31) = 255.255.255.252

Sehingga pembagian *subnetting*-nya adalah:

Tabel 2. Subnetting Router

Subnet	Antarmuka 1	Antarmuka 2	Subnet Mask
Router 1	10.10.10.1/30	10.10.20.2/30	255.255.255.252
Router 2	10.10.20.1/30	10.10.30.2/30	255.255.255.252
Router 3	10.10.30.1/30	10.10.40.2/30	255.255.255.252
Router 4	10.10.40.1/30	10.10.50.2/30	255.255.255.252
Router 5	10.10.50.1/30	10.10.60.2/30	255.255.255.252
Router 6	10.10.60.1/30	10.10.10.2/30	255.255.255.252

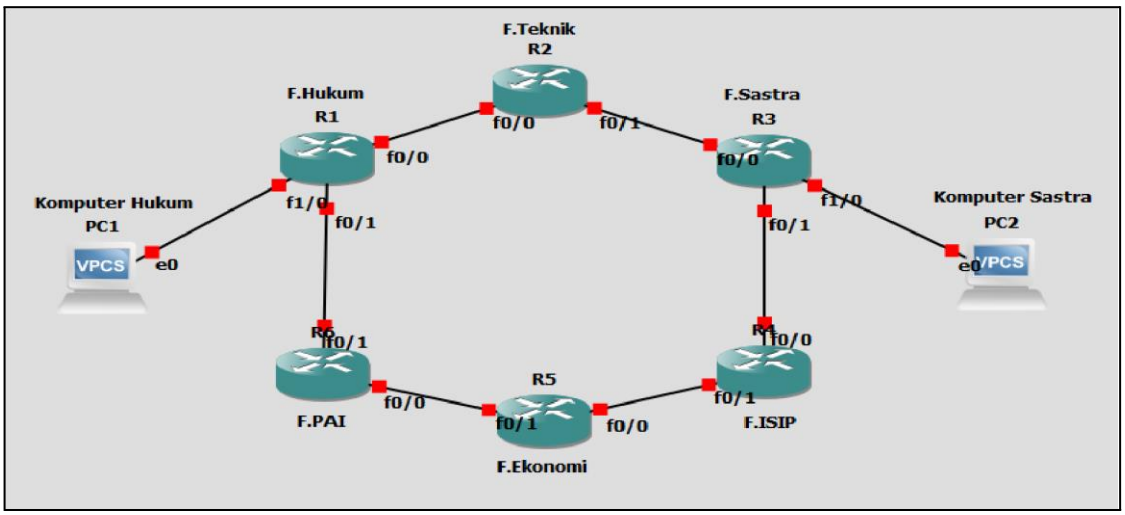
Kemudian untuk *subnetting* antar PC dan *router* yang pada penelitian ini terdapat 2 buah PC digunakan alamat ip digunakan alamat ip dengan kelas b dan c dengan panjang *prefix* /24 yang dapat digunakan untuk 254 host. *Subnet mask* untuk panjang *prefix* /24 adalah 255.255.255.0 yang ditampilkan pada tabel berikut.

Tabel 3. Tabel Subnetting Router

Subnet	IP	Subnet Mask	Gateaway
PC 1	192.168.1.1	255.255.255.0	192.168.1.254
PC 2	172.16.10.1	255.255.255.0	172.16.10.254
Router 1	192.168.1.254	255.255.255.0	-
Router 2	172.16.10.254	255.255.255.0	-

Untuk PC dan *router* memiliki *gateway* sebagai pintu masuk untuk jaringan yang berbeda. Dalam hal ini hubungan antara *router* tidak memiliki *gateway* karena pengaturan tersebut akan diserahkan pada metode OSPF nya.

Pada penelitian ini simulasi di inisiasikan pada 6 fakultas yang mewakili masing-masing router berikut merupakan skema rancangan topologi yang disusun pada penelitian ini.



Gambar 3. Skema Rancangan Topologi

Pada gambar diatas dapat dilihat bahwa *router* mewakili setiap fakultas yang disimulasikan dan pada router 1 dan 3 terdapat hubungan antara *router* dan pc yang nantinya akan menjadi alat untuk melakukan pengujian pengiriman paket. Untuk lebih jelas mengenai pengalokasian alamat pada masing-masing perangkat berikut tabel penyempurnaan pada penetapan alamat ip yang digunakan.

Tabel 4. Tabel Addressing

Perangkat	Antarmuka	Alamat IP	Subnet Mask	Gateway
R1/ Router F. Hukum	Fa 0/0	10.10.10.1	255.255.255.252	N/A
	Fa 0/1	192.168.1.254	255.255.255.0	N/A
	Fa 1/0	10.10.60.2	255.255.255.252	N/A
R2/ Router F. Teknik	Fa 0/0	10.10.10.2	255.255.255.252	N/A
	Fa 0/1	10.10.20.1	255.255.255.252	N/A
R3/ Router F. Sastra	Fa 0/0	10.10.20.2	255.255.255.252	N/A
	Fa 0/1	172.16.10.254	255.255.255.0	N/A
	Fa 1/0	10.10.50.2	255.255.255.252	N/A
R4/ Router F. PAI	Fa 0/0	10.10.30.2	255.255.255.252	N/A
	Fa 0/1	10.10.50.1	255.255.255.252	N/A
R5/ Router F. ISIP	Fa 0/0	10.10.40.2	255.255.255.252	N/A
	Fa 0/1	10.10.50.2	255.255.255.252	N/A
R6/ Router F. ISIP	Fa 0/0	10.10.60.1	255.255.255.252	N/A
	Fa 0/1	10.10.50.2	255.255.255.252	N/A
PC0/ Komputer Hukum	NIC	192.168.1.1	255.255.255.0	192.168.1.254
PC1/ Komputer Sastra	NIC	172.16.10.1	255.255.255.0	172.16.10.254

Untuk melakukan konfigurasi OSPF kita menggunakan jalur terminal sebuah *router* dimana pada penelitian ini router yang digunakan adalah *router Cisco* 3725 dengan jumlah *default interface* sebanyak 3 buah yang akan menjadi jalur penghubung antar *router*. Berikut merupakan tabel konfigurasi untuk implementasi OSPF pada setiap *router*.

 https://doi.org/10.56211/sudo.v3i2.532

Tabel 5. Konfigurasi OSPF Router

No.	*Device*	Konfigurasi
1	Router Hukum	*Router>enable* *Router#configure terminal* *Router(config)#router ospf 1* *Router(config-router)#router-id 1.1.1.1* *Router(config-router)#network 10.10.10.0 0.0.0.3 area 0* *Router(config-router)#network 10.10.60.0 0.0.0.3 area 0* *Router(config-router)#network 192.168.1.0 0.0.0.255 area 0* *Router(config-router)#no shutdown* *Router(config-router)#exit* *Router(config)#exit* *Router#write memory*
2	Router Teknik	*Router>enable* *Router#configure terminal* *Router(config)#router ospf 1* *Router(config-router)#router-id 2.2.2.2* *Router(config-router)#network 10.10.10.0 0.0.0.3 area 0* *Router(config-router)#network 10.10.20.0 0.0.0.3 area 0* *Router(config-router)#no shutdown* *Router(config-router)#exit* *Router(config)#exit* *Router#write memory*
3	Router Sastra	*Router>enable* *Router#configure terminal* *Router(config)#router ospf 1* *Router(config-router)#router-id 3.3.3.3* *Router(config-router)#network 10.10.20.0 0.0.0.3 area 0* *Router(config-router)#network 10.10.30.0 0.0.0.3 area 0* *Router(config-router)#network 172.16.10.0 0.0.0.255 area 0* *Router(config-router)#no shutdown* *Router(config-router)#exit* *Router(config)#exit* *Router#write memory*
4	Router Fisip	*Router>enable* *Router#configure terminal* *Router(config)#router ospf 1* *Router(config-router)#router-id 4.4.4.4* *Router(config-router)#network 10.10.30.0 0.0.0.3 area 0* *Router(config-router)#network 10.10.40.0 0.0.0.3 area 0* *Router(config-router)#no shutdown* *Router(config-router)#exit* *Router(config)#exit* *Router#write memory*
5	Router Ekonomi	*Router>enable* *Router#configure terminal* *Router(config)#router ospf 1* *Router(config-router)#router-id 5.5.5.5* *Router(config-router)#network 10.10.50.0 0.0.0.3 area 0* *Router(config-router)#network 10.10.40.0 0.0.0.3 area 0*

		Router(config-router)#no shutdown *Router(config-router)#exit* *Router(config)#exit* *Router#write memory*
6	Router PAI	*Router>enable* *Router#configure terminal* *Router(config)#router ospf 1* *Router(config-router)#router-id 6.6.6.6* *Router(config-router)#network 10.10.50.0 0.0.0.3 area 0* *Router(config-router)#network 10.10.60.0 0.0.0.3 area 0* *Router(config-router)#no shutdown* *Router(config-router)#exit* *Router(config)#exit* *Router#write memory*

Validasi Konfigurasi OSPF

Setelah terkonfigurasi maka dilakukan pengujian koneksi antara PC dan interface router tersebut seperti yang ditampilkan pada gambar dibawah ini.

```
PC1> ping 192.168.1.254

84 bytes from 192.168.1.254 icmp_seq=1 ttl=255 time=10.653 ms
84 bytes from 192.168.1.254 icmp_seq=2 ttl=255 time=6.949 ms
84 bytes from 192.168.1.254 icmp_seq=3 ttl=255 time=9.469 ms
84 bytes from 192.168.1.254 icmp_seq=4 ttl=255 time=4.771 ms
84 bytes from 192.168.1.254 icmp_seq=5 ttl=255 time=1.126 ms
```

Gambar 4. Ping Test PC1

```
PC2> ping 172.16.10.254

84 bytes from 172.16.10.254 icmp_seq=1 ttl=255 time=10.446 ms
84 bytes from 172.16.10.254 icmp_seq=2 ttl=255 time=6.930 ms
84 bytes from 172.16.10.254 icmp_seq=3 ttl=255 time=6.233 ms
84 bytes from 172.16.10.254 icmp_seq=4 ttl=255 time=7.092 ms
84 bytes from 172.16.10.254 icmp_seq=5 ttl=255 time=7.207 ms
```

Gambar 5. Ping Test PC2

Dari kedua gambar diatas dapat dilihat bahwa pengujian konektifitas pada router dan PC telah berjalan dengan baik pada kedua PC. Oleh karena itu maka keseluruhan jaringan OSPF pada topologi yang diusulkan telah berjalan dengan baik dan benar.

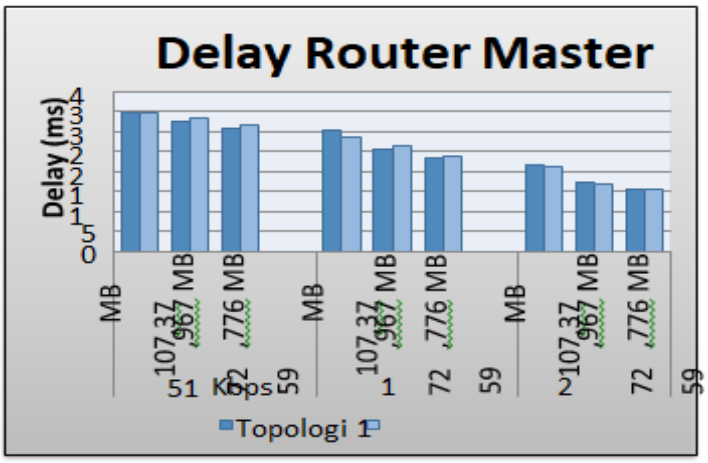
Analisis Packet Loss, Throughput dan Delay

Pada tahap analisis *packet loss, throughput, delay* ini dilakukan dengan melakukan monitoring menggunakan aplikasi *wireshark* dimana aplikasi monitor jaringan ini sudah diintegrasikan dengan GNS3 sehingga memungkinkan untuk mampu menangkap lalu lintas yang terjadi pada jaringan yang telah dibuat sebelumnya. Berikut analisis yang dilakukan pada penelitian ini.

 https://doi.org/10.56211/sudo.v3i2.532

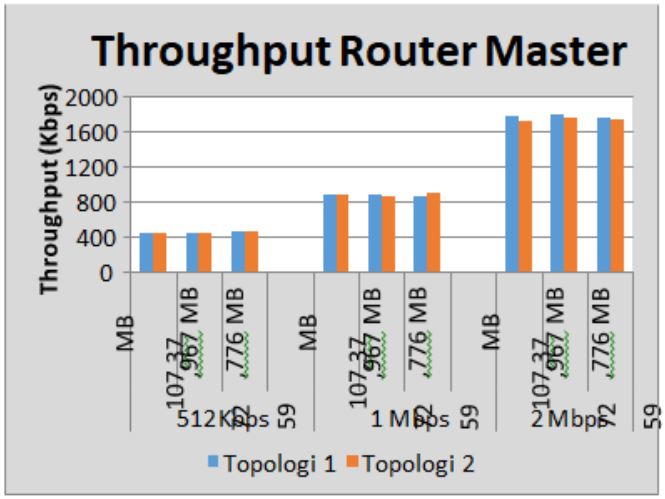
Gambar 6. *Grafik Packet loss*

Throughput mengukur jumlah data yang berhasil dikirim dari satu titik ke titik lain dalam satuan waktu tertentu, biasanya dalam bit per detik (bps), kilobit per detik (kbps), atau megabit per detik (Mbps). *Delay*, atau *Latency*, adalah waktu yang dibutuhkan untuk sebuah paket data melakukan perjalanan dari pengirim ke penerima. Ini adalah metrik penting dalam mengukur kinerja jaringan. Terlihat pada tabel grafik delay paling rendah pada test 1 yaitu 2,058.



Gambar 7. Grafik *Delay*

Delay pada *router master* merupakan kombinasi dari berbagai komponen yang mempengaruhi waktu keseluruhan yang dibutuhkan untuk mengirim paket data melalui *router*. Memahami dan mengelola faktor-faktor yang mempengaruhi *delay* ini penting untuk memastikan kinerja jaringan yang optimal dan meminimalkan *latency* bagi pengguna.



Gambar 8. Grafik *Troughput*

Throughput pada *router master* adalah ukuran penting dari seberapa banyak data yang dapat diproses dan diteruskan oleh *router* dalam satuan waktu tertentu. Dengan memahami faktor-faktor yang mempengaruhi *throughput* dan menerapkan *strategi* untuk meningkatkannya, *administrator* jaringan dapat memastikan bahwa router master berfungsi dengan efisien dan memenuhi kebutuhan jaringan yang terus berkembang.

Analisis Packet Loss

Packet loss merupakan kondisi dimana data yang dikirim melalui jaringan tidak berhasil tiba di tujuan. Hal ini dapat menurunkan kualitas layanan serta pengalaman pada pengguna jaringan. Untuk menganalisis *packet loss* maka dilakukan skenario pengiriman *Ping tcp* pada GNS3 kemudian menangkapnya dengan aplikasi *wireshark*. Untuk melakukan pengiriman *Ping TCP* pada aplikasi GNS3 dilakukan dengan cara memberikan perintah *Ping* pada PC1 dengan alamat ip 192.168.1.1 menuju 172.16.10.1 dengan mode TCP. Berikut merupakan gambar dari perintah tersebut.

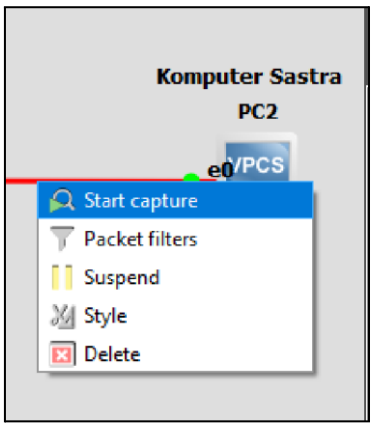
```
PC1> ping 172.16.10.1 -3

Connect   7@172.16.10.1 seq=1 ttl=61 time=2062.714 ms
SendData  7@172.16.10.1 seq=1 ttl=61 time=45.074 ms
Close     7@172.16.10.1 seq=1 ttl=61 time=44.701 ms
Connect   7@172.16.10.1 seq=2 ttl=61 time=53.198 ms
SendData  7@172.16.10.1 seq=2 ttl=61 time=50.314 ms
Close     7@172.16.10.1 seq=2 ttl=61 time=61.433 ms
Connect   7@172.16.10.1 seq=3 ttl=61 time=54.535 ms
SendData  7@172.16.10.1 seq=3 ttl=61 time=51.228 ms
Close     7@172.16.10.1 seq=3 ttl=61 time=59.734 ms
Connect   7@172.16.10.1 seq=4 ttl=61 time=42.414 ms
SendData  7@172.16.10.1 seq=4 ttl=61 time=47.230 ms
Close     7@172.16.10.1 seq=4 ttl=61 time=55.623 ms
Connect   7@172.16.10.1 seq=5 ttl=61 time=47.220 ms
SendData  7@172.16.10.1 seq=5 ttl=61 time=43.722 ms
Close     7@172.16.10.1 seq=5 ttl=61 time=53.699 ms
```

Gambar 9. TCP Ping Test PC1

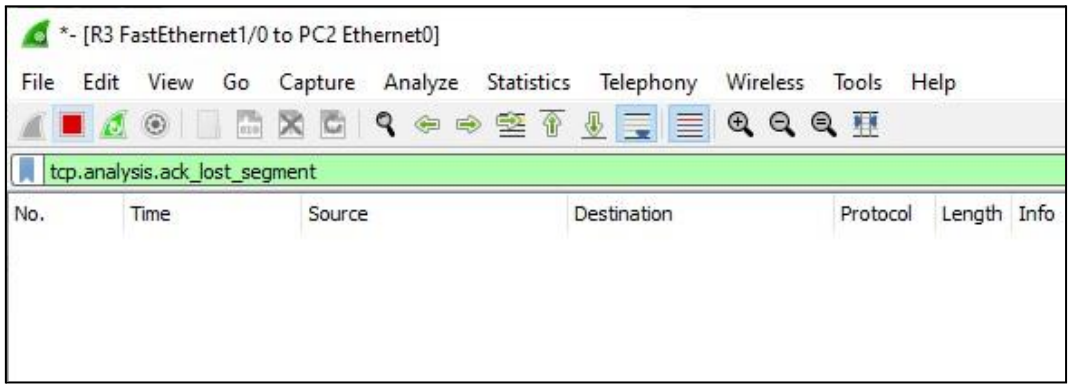
Dari gambar diatas dapat diketahui bahwa dengan melakukan perintah ping 172.16.10.1 -3 pada termina virtual PC akan secara otomatis mengirimkan pesan acak menuju pc dengan ip tersebut. Dari gambar tersebut seluruh komunikasi berjalan dengan lancar dan tidak terdapat paket yang hilang atau pun terjadi RTO (*request time out*). Kemudian pada aplikasi GNS3 kita lakukan proses *capture* dengan cara seperti gambar dibawah.

 https://doi.org/10.56211/sudo.v3i2.532



Gambar 10. *Start Capture*

Dengan melakukan "*start capture"* seperti gambar diatas maka sejalan dengan itu aplikasi monitoring jaringan yaitu wireshark akan berjalan dan kita melakukan melakukan *capture* filtering dengan filter "*tcp.analysis.lost.segment*" untuk mengetahui apakah terdapat paket yang hilang atau gagal terkirim. Berikut hasil *capture* pada aplikasi *wireshark* tersebut.



Gambar 11. *Filter Packet Loss*

Dari gambar diatas tidak ditemui adanya paket yang hilang yang tertangkap oleh aplikasi *wireshark*. Oleh karena itu maka diperoleh kesimpulan bahwa *packet loss* adalah 0% pada skenario yang telah dilakukan. Dengan demikian maka rangkuman untuk perhitungan *packet loss* sesuai dengan rangkuman pada *statistic* dengan filter "tcp" *wireshark* sebagaimana tertera dibawah.

Statistics

Measurement	Captured	Displayed	Marked
Packets	849	47 (5.5%)	—
Time span, s	3581.261	8.486	—
Average pps	0.2	5.5	—
Average packet size, B	94	68	—
Bytes	79921	3178 (4.0%)	0
Average bytes/s	22	374	—
Average bits/s	178	2996	—

Gambar 12. Statistic Packets

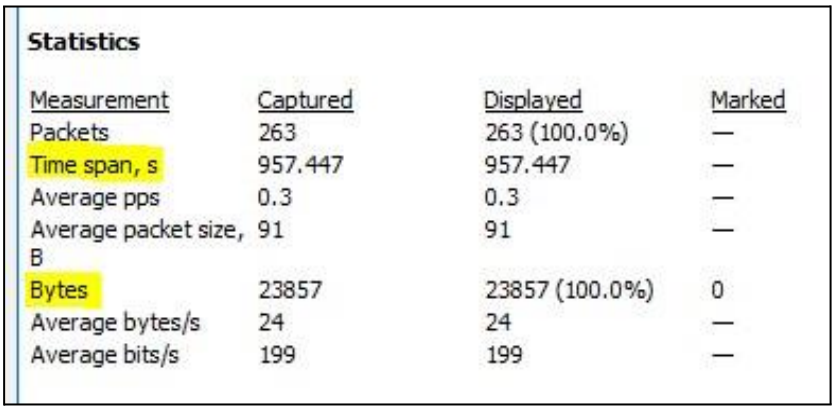
Dari gambar diatas dapat dilihat pada *statistic wireshark* pada nilai "*Packets*" memiliki nilai 849 dikarenakan paket terkirim dan diterima bernilai 100% maka *packet loss* pada skenario ini adalah 0%

Analisis Throughput

Untuk menghitung nilai *throughput* maka diperlukan nilai total data yang diterima dan lama pengamatan atau waktu pengiriman sesuai dengan rumus dibawah ini.

$$Throughput(byte) = \frac{TotalDataYangDiterima}{LamaPengamatan}$$

Kedua nilai tersebut dapat diperoleh dari rangkuman statistik yang telah diperoleh seperti gambar dibawah ini:



Statistics

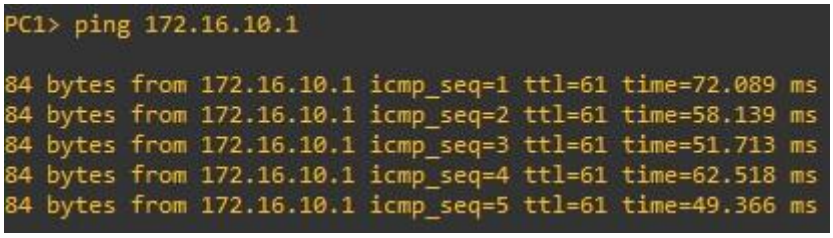
Measurement	Captured	Displayed	Marked
Packets	263	263 (100.0%)	—
Time span, s	957.447	957.447	—
Average pps	0.3	0.3	—
Average packet size, B	91	91	—
Bytes	23857	23857 (100.0%)	0
Average bytes/s	24	24	—
Average bits/s	199	199	—

Gambar 13. Statistic Throughput

Perhitungan untuk nilai *throughput* adalah sebagai berikut:
Throughput = 3581^{79921},261 = 22,3164 bytes/s

Analisis Delay

Untuk mengukur *delay* dilakukan skenario pengiriman ping dengan mode ICMP (*Internet Control Message Protocol*) untuk mempermudah melihat waktu pengiriman dan waktu penerimaan yang terjadi pada saat komunikasi. Untuk melakukan hal tersebut berikut gambar pengiriman ping pada PC1 menuju PC2.

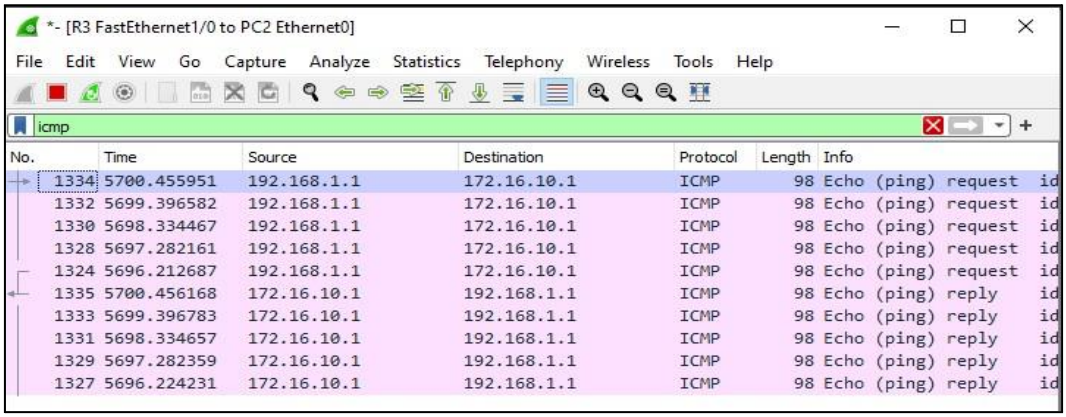


```
PC1> ping 172.16.10.1

84 bytes from 172.16.10.1 icmp_seq=1 ttl=61 time=72.089 ms
84 bytes from 172.16.10.1 icmp_seq=2 ttl=61 time=58.139 ms
84 bytes from 172.16.10.1 icmp_seq=3 ttl=61 time=51.713 ms
84 bytes from 172.16.10.1 icmp_seq=4 ttl=61 time=62.518 ms
84 bytes from 172.16.10.1 icmp_seq=5 ttl=61 time=49.366 ms
```

Gambar 14. ICMP Ping Test

Pada gambar diatas dapat dilihat bahwa pengiriman berhasil dilakukan tanpa RTO (*request time out*) ataupun kehilangan paket pada ip tujuan 172.16.10.1. Setelah melakukan pengiriman maka kita melakukan *filtering* pada aplikasi wireshark dengan filter "icmp" untuk menyaring seluruh komunikasi dengan ciri tersebut. Berikut gambar hasil *filtering* yang dilakukan pada aplikasi monitoring jaringan *wireshark*.



*- [R3 FastEthernet1/0 to PC2 Ethernet0]

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

icmp

No.	Time	Source	Destination	Protocol	Length	Info
1334	5700.455951	192.168.1.1	172.16.10.1	ICMP	98	Echo (ping) request id
1332	5699.396582	192.168.1.1	172.16.10.1	ICMP	98	Echo (ping) request id
1330	5698.334467	192.168.1.1	172.16.10.1	ICMP	98	Echo (ping) request id
1328	5697.282161	192.168.1.1	172.16.10.1	ICMP	98	Echo (ping) request id
1324	5696.212687	192.168.1.1	172.16.10.1	ICMP	98	Echo (ping) request id
1335	5700.456168	172.16.10.1	192.168.1.1	ICMP	98	Echo (ping) reply id
1333	5699.396783	172.16.10.1	192.168.1.1	ICMP	98	Echo (ping) reply id
1331	5698.334657	172.16.10.1	192.168.1.1	ICMP	98	Echo (ping) reply id
1329	5697.282359	172.16.10.1	192.168.1.1	ICMP	98	Echo (ping) reply id
1327	5696.224231	172.16.10.1	192.168.1.1	ICMP	98	Echo (ping) reply id

Gambar 15. ICMP Ping Test

 https://doi.org/10.56211/sudo.v3i2.532

Pada gambar diatas filtering berhasil dilakukan dan mendapatkan hasil komunikasi yang telah dilakukan sebelumnya. Untuk memperoleh nilai *delay* dari komunikasi tersebut maka kita harus mengolah nilai waktu (*time*) yang terdapat pada saat mengirim dan menerima dimana hal tersebut dapat diketahui dari jenis info *request* dan *reply* pada data diatas. Berikut tabulasi dari data berdasarkan rumus delay berikut.

$$Delay(ms) = WaktuPaketDiterima - WaktuPaketDikirim$$

Tabel 6. Waktu Delay

No	Waktu Terima	Waktu Kirim
1	5696,224231	5696,212687
2	5.697,282359	5697,282161
3	5.698,334657	5698,334467
4	5699,396783	5699,396582
5	5700,456168	5700,455951
Total	**28491.6942**	**28491.68185**

Perumusan pada tabel delay diatas sebagai berikut ini:
Waktu Terima – Waktu Kirim

1. 5696,224231- 5696,212687 = 0.01235
2. 5.697,282359 - 5697,282161 = 0.002466
3. 5.698,334657 - 5698,334467 = 0.011854
4. 5699,396783 - 5699,396582 = 0.001217
5. 5700,456168 - 5700,455951 = 0.012632

Dari hasil perhitungan pada tabel diatas maka dapat dilihat bahwa *delay* yang diperoleh sebesar 0.01235 ms. Nilai dari *delay* tersebut terindikasi sebagai kategori pada jaringan yang sangat bagus berdasarkan kategori di bawah ini.

Tabel 7. *Indeks Karakteristik Delay*

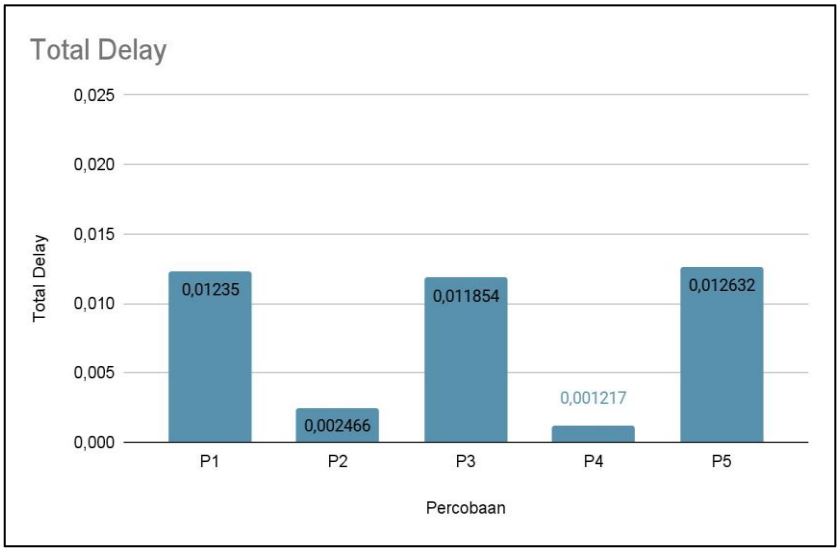
Kategori	Besar Delay (ms)	Indeks
Sangat Bagus	<150	4
Bagus	150 - 300	3
Sedang	300 - 450	2
Jelek	>450	1

Untuk memberikan pengamatan tentang kestabilan *delay* maka dilakukan simulasi dengan skenario yang sama seperti sebelumnya sebanyak 5 kali. Berikut tabulasi dari percobaan yang dilakukan.

Tabel 8. Percobaan *Delay*

Percobaan	Total Delay
1	0.01235
2	0.002466
3	0.011854
4	0.001217
5	0.012632

Tabel diatas merupakan nilai dari total *delay* dari seluruh percobaan yang dilakukan pada penelitian ini untuk menguji besar *delay* dengan skenario yang sama seperti yang telah dilakukan diatas. Untuk lebih memperjelas hasil dari *delay* tersebut maka ditampilkan diagram dibawah ini.



Gambar 16. Gambar ICMP *Ping Test*

Dari diagram diatas dapat diketahui bahwa seluruh total *delay* dari percobaan menunjukkan angka yang sangat rendah dalam hal ini sangat bagus untuk kategori jaringan. Nilai-nilai yang diperoleh memiliki nilai yang jauh bahkan lebih rendah dari nilai maksimum pada diagram yaitu 0.25 dimana dari hal tersebut dapat disimpulkan bahwa lalu lintas komunikasi jaringan berjalan dengan sangat baik dan juga hasil dari *packet loss* yang diperoleh dari percobaan sebelumnya juga menunjukkan bahwa seluruh paket terkirim dan tidak terjadi kehilangan dapat merugikan pengguna jaringan. Oleh karena itu dapat disimpulkan bahwa penggunaan OSPF pada penelitian ini berjalan dengan lancar pada seluruh elemen yang dikonfigurasikan.

KESIMPULAN DAN SARAN

Untuk mengukur dan menganalisis *delay*, *throughput* dan *packet loss* dalam jaringan UISU menggunakan alat bantu pengukuran yang disediakan oleh *Graphical Network Simulator*. Menganalisis *packet loss* maka dilakukan skenario pengiriman *ping tcp* pada GNS3 kemudian menangkapnya dengan aplikasi *wireshark* dan analisis *packet loss, throughput, delay* ini dilakukan dengan melakukan monitoring menggunakan aplikasi *wireshark* dimana aplikasi monitor jaringan ini sudah diintegrasikan dengan GNS3 sehingga memungkinkan untuk mampu menangkap lalu lintas yang terjadi pada jaringan.

DAFTAR PUSTAKA

[1] M. Jannah and A. Basuki, “Visualisasi Topologi Routing pada Jaringan berdasarkan OSPF Link State Database,” vol. 7, no. 3, pp. 1329–1335, 2023, [Online]. Available: http://j-ptiik.ub.ac.id

[2] D. A. RetnaniWulandari, Y. A. Auliya, A. CahyaPrihandoko, S. Slamin, and M. Zarkasi, “Wireless Area Network Infrastructure Model on Gili Ketapang Island Using Open Shortest Path First Routing Protocol,” Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control, Feb. 2022, doi: 10.22219/kinetik.v7i1.1373.

[3] N. Iryani, A. D. Ramadhani, and M. K. Sari, “Analisis Performansi Routing OSPF menggunakan RYU Controller dan POX Controller pada Software Defined Networking,” J. Telekomun. Dan Komput., vol. 11, no. 1, p. 73, Apr. 2021, doi: 10.22441/incomtech.v11i1.10187.

[4] R. Artha and B. Soewito, “Perancangan Ulang Topologi Jaringan Dengan Kerangka Kerja Ppdioo Network Topology Redesign With Ppdioo Framework,” vol. 13, no. 1, pp. 34–41, 2023.

[5] Rengel Julian and Alek Wijaya, “Analisa dan pengembangan Jaringan WAN Pada Gedung Bagian Lalin Di Dishub Pemprov Sumsel,” Semin. Has. Penelit. Vokasi, pp. 34–40, 2017.

[6] S. Alvionita and H. Nurwasito, "Analisis Kinerja Protokol Routing OSPF, RIP dan EIGRP Pada Topologi Jaringan Mesh," 2019. [Online]. Available:http://j-ptiik.ub.ac.id

[7] T. M. Diansyah, D. Handoko, I. Faisal, A. Yunianti, K. Chiuloto, and R. Liza, "Design Analysis of OSPF (Open Shortest Path First) Routing by Calculating Packet Loss of Network WAN (Wide Area Network)," in Journal of Physics: Conference Series, Institute of Physics Publishing, Dec. 2019. doi: 10.1088/1742-6596/1361/1/012087.

[8] Y. Rahmawati and N. Mutiara Anjani, "Implementation of Link Failover on Metronet Network PT. Telkom Indonesia (Persero) Based on Ipv4 andOSPF," J. INFORMATICS Telecommun. Eng., vol. 6, no. 2, pp. 458–472, Jan. 2023, doi: 10.31289/jite.v6i2.8313.