JURNAL SISFOTENIKA
Vol. 12 No. 1, Januari 2022
p-ISSN: 2087-7897 ; e-ISSN : 2460-5344
DOI : 10.30700/jst.v12i1.1221

Penerapan Metode *Confederation*, *Route reflector* Dan *Next hop self* Pada *Routing* BGP

Application Of Confederation Methods, Route reflector And Next hop self On BGP Routing

Gilfan Niando*[1], Iwan Wahyuddin[2], Andrianingsih[3]
[1,2]Universitas Nasional; Jl. Sawo Manila, RT.14/RW.3, Pejaten Bar., Kec. Ps. Minggu, Kota Jakarta Selatan, DKI Jakarta 12520, (021) 7806700
[3]Jurusan Informatika, FTKI UNAS, DKI Jakarta
e-mail: *[1]gniando@gmail.com, [2]iwan_wyd@yahoo.com, [3]andrianingsih@civitas.unas.ac.id

Abstrak

Kemajuan teknologi komunikasi mempengaruhi kebutuhan akan jaringan yang lebih besar. Semakin besar jaringan, semakin banyak rute yang ada. Di BGP, semakin besar sesi yang terbentuk pada AS, semakin lambat komunikasi jaringan. Beberapa cara untuk mengurangi jumlah informasi di tabel routing adalah menggunakan metode confederation , route reflector dan route next hop self. Dalam penelitian ini akan membahas simulasi penerapan metode confederation , route reflector dan route next hop self pada routing BGP pada hubungan antar AS yang berbeda untuk mengurangi jumlah informasi di tabel routing. Penelitian ini dibuat untuk memberikan solusi syarat full-mesh pada BGP. Penerapannya menggunakan software GNS3 untuk membuat prototipe dan simulasi jaringan. Sehingga dapat diterapkan pada jaringan nyata nantinya. Pengujian yang dilakukan adalah menghitung forwarding delay , delay dan throughput. Berdasarkan percobaan yang dilakukan, ketiga metode tersebut menghasilkan 7% lebih rendah pada forwarding delay, 29% lebih rendah pada end to end delay, dan 26% lebih tinggi pada throughput dibandingkan BGP full-mesh. Hal ini membuktikan ketiga metode ini dapat mengurangi tabel routing dan membuat pengiriman packet data menjadi lebih cepat. Untuk penelitian selanjutnya diharapkan dapat diterapkan pada studi kasus nyata sehingga dapat dinilai efektifitas dan efisiensi serta dampak konfigurasi yang digunakan secara nyata.

Kata kunci*— BGP, Confederation, Next Hop, Reflector.*

Abstract

Advances in communication technology affect the need for larger networks. The bigger the network, the more routes there are. In BGP, the larger the session formed on the AS, the slower the network communication. Some ways to reduce the amount of information in the routing table are to use the confederation method, route reflector and route next hop self. In this study, we will discuss the simulation of the application of the confederation method, route reflector and route next hop self in BGP routing on different AS relationships to reduce the amount of information in the routing table. This study was made to provide a solution for full-mesh conditions in BGP. The application uses GNS3 software to create prototypes and network simulations. So that it can be applied to real networks later. The test is to calculate the

forwarding delay, delay and throughput. Based on the experiments conducted, the three methods resulted in 7% lower forwarding delay, 29% lower end-to-end delay, and 26% higher throughput than full-mesh BGP. This proves that these three methods can reduce the routing table and make packet data delivery faster. For further research, it is hoped that it can be applied to real case studies so that effectiveness and efficiency can be assessed as well as the impact of the configuration used in real.

Keywords— *BGP, Confederation, Next Hop, Reflector.*

1. PENDAHULUAN

Internet adalah kumpulan dari komputer yang terhubung secara fisik dalam sebuah jaringan. Internet membawa lalu lintas dalam bentuk informasi yang dikirim dan diterima oleh komputer atau mesin di dua lokasi yang berbeda [1]. Dalam beberapa tahun terakhir, jumlah pengguna internet semakin meningkat pesat dan jumlah penggunanya hingga kini terus bertambah. Hal ini dapat dilihat dari angka *Autonomous System* (AS) yang terpakai lebih dari 35000 di tahun 2011 [2]. Secara administratif, internet dibagi menjadi ribuan AS yang bertukar informasi dengan *routing external* BGP (*Border Gateway Protocol*). BGP dibagi menjadi dua yaitu iBGP dan eBGP, iBGP dikhususkan untuk menghubungkan satu AS (internal). Sedangkan eBGP dikhususkan untuk menghubungkan antar AS yang berbeda (eksternal).

AS yang dibangun menggunakan *routing* iBGP harus terhubung ke semua sistem BGP lainnya, sehingga membentuk konfigurasi *full-mesh*. Ini menjadi perhatian karena jumlah pengguna internet di masing-masing wilayah terus bertambah. Maka penggunaan *router* pada satu jaringan besar AS secara tidak langsung juga meningkat, sehingga meningkatkan informasi *routing* dalam tabel *routing* BGP [3], [4]. Setiap *router* iBGP mengirimkan informasi *next hop* ke tetangganya, maka setiap *router* di jaringan AS harus memiliki informasi dari setiap tetangga. Sehingga proses pengiriman data memerlukan proses pencarian yang jauh lebih lama daripada konfigurasi non-*fullmesh*. Hal ini memungkinkan CPU dan memory terbebani untuk menangani data yang lewat dari satu *router* ke next hop *router*nya. Untuk mengatasinya dapat menggunakan metode *confederation* dengan membagi satu AS internal menjadi beberapa subAS, metode *route reflector* dengan membuat satu *router* menjadi *reflector* yang akan memantulkan ke semua *router* iBGP lainnya, dan metode *next hop self* dengan tujuan agar *gateway* dari sebuah *router* untuk melewati *router* yang mengaktifkan *next hop self* ini akan digunakan sebagai *gateway* [5]. Sehingga dapat mempercepat pencarian pada tabel *routing* dengan mengurangi jumlah tabel *routing* yang terbentuk akibat full-mesh.

Oleh karena itu dalam penelitian ini disimulasikan suatu penerapan metode *confederation* , metode *route reflector* dan metode *next hop self* untuk mengatasi permasalahan *fullmesh* dengan mengambil kasus transmisi data *File Transfer Protocol (FTP)* dalam jaringan AS tersebut.

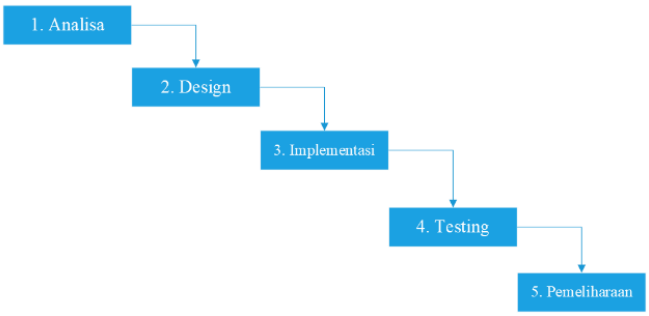
Penelitan sebelumnya yang berhubungan dengan penelitian ini adalah Hari Antoni Musril pada “Simulasi Interkoneksi Antara Autonomous System Menggunakan Border Gateway Protocol (BGP)” Tahun 2017 menjelaskan bahwa BGP bertindak sebagai protokol *routing* antara sistem otonom yang berbeda. Dengan demikian, ia dapat bertukar rute masuk dan keluar dari jaringan lokal dengan nomor sistem otonom. BGP juga merupakan protokol perutean yang sangat andal. Ini karena BGP menggunakan TCP untuk berkomunikasi dengan tetangganya saat bertukar informasi [6]. Penelitian kedua yang memiliki relevansi dengan penelitian ini adalah “Analisis Performansi Jaringan Autonomous System Dengan Metode *Confederation*” oleh Erwan Ramdhani Saputra Mahasiswa Prodi Teknik Informatika Institut Teknologi Telkom,

Bandung Tahun 2012. Penelitian tersebut menjelaskan tentang penggunaan metode *confederation* untuk menyelesaikan masalah full-mesh [1].

2. METODE PENELITIAN

2.1 Metode Waterfall

Pada penelitian ini menggunakan metode waterfall yang menggambarkan suatu proses lengkap, diantaranya seperti pada Gambar 1.



Gambar 1. Model Metode Waterfall

a. Analisis. Pada tahap ini dilakukan analisis terhadap literatur yang relevan. Literatur berasal dari buku, jurnal ilmiah, dan penelitian terkait Border Gateway Protocol (BGP). Pada tahap ini juga dilakukan tentang aturan *fullmesh* yang dipakai dalam jaringan.
b. Desain. Tahap ini melibatkan bentuk prototipe yang dikembangkan dari topologi jaringan, membahas skema jaringan dan menerapkan metode *confederation* , metode *route reflector* dan metode *next hop self* untuk memecahkan masalah *fullmesh*. Pengembangan prototype jaringan menggunakan software simulasi jaringan komputer Graphical Network Simulator 3.
c. Implementasi. Tahap selanjutnya mengkonfigurasi prototipe topologi jaringan. Konfigurasi dilakukan pada semua perangkat di jaringan prototipe, , termasuk komputer client , komputer server , dan *router*. Protokol perutean BGP dikonfigurasi pada *router* dengan memasukkan kode pemrograman ke jendela CLI *router*. *Router* dikonfigurasi sedemikian rupa sehingga menyediakan jalur terbaik untuk mentransmisikan paket data di jaringan.
d. Testing. Setelah mengembangkan jaringan prototipe. Setiap perangkat diuji konektivitasnya. Kemudian melakukan pemeriksaan *routing* BGP.
e. Pemeliharaan, langkah terakhir dalam metode ini adalah memulai sistem dan melakukan perawatan untuk memperbaiki kesalahan sistem yang tidak terdeteksi pada langkah sebelumnya.

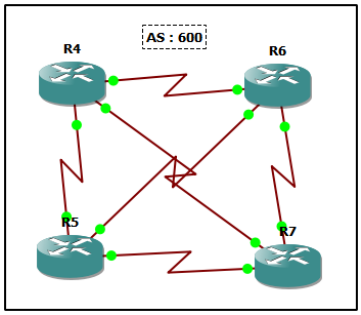
2.2. Border Gateway Protocol

BGP dibagi menjadi dua yaitu iBGP dan eBGP, iBGP dikhususkan untuk menghubungkan satu AS (internal). Sedangkan eBGP dikhususkan untuk menghubungkan antar AS yang berbeda (eksternal). Dalam iBGP memiliki beberapa aturan salah satunya adalah *fullmesh*. Hal ini dikarenakan BGP memiliki aturan split horizon dimana saat suatu router menerima update dari router iBGP lain, maka router tersebut tidak akan memforward paket tersebut ke router iBGP lain. Split horizon digunakan untuk mencegah routing loop.

BGP memiliki kemampuan melakukan pengumpulan rute, pertukaran rute dan menentukan rute terbaik menuju ke sebuah lokasi dalam jaringan [7]. BGP sangat terukur karena dapat menangani pertukaran perutean di banyak organisasi besar. BGP berfokus pada pemilihan rute antar domain dan antar nomor AS di Internet.

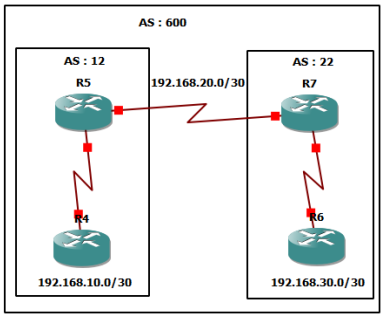
2.2.1. Metode Confederation

Semakin besar sebuah jaringan maka semakin banyak pula rute yang ada di dalam nya. Pada BGP ini berakibat pada semakin besar nya sesi yang terbentuk dalam AS. BGP *confederation* adalah suatu metode untuk menangani masalah di atas. Menurut [8] BGP *confederation* adalah mekanisme yang terdapat pada BGP untuk membagi satu AS menjadi beberapa AS yang lebih kecil sehingga dapat memperkecil rentang path AS serta mengurangi kompleksitas sesi BGP yang terbentuk . Dari luar AS, BGP *confederation* terlihat sebagai satu AS meskipun di dalam AS yang menerapkan *confederation* tersebut terbagi lagi menjadi beberapa AS yang lebih kecil, AS yang menerapkan *confederation* tetap menggunakan satu buah ASN sebagai indetifier-nya. Penggabungan AS ini mempermudah dalam pengaturan policy dan traffic engineering.



Gambar 2. Topologi *Fullmesh*

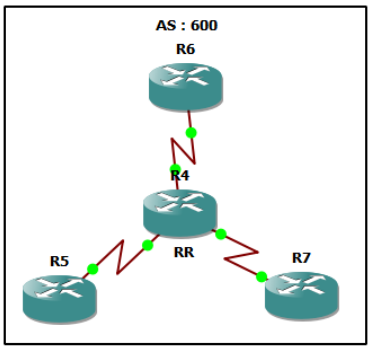
Pada Gambar 2 terlihat jaringan full mesh yang mengakibatkan banyak rute yang ada di dalamnya dan berakibat semakin besarnya sesi yang terbentuk. Maka dapat diaplikasi metode *confederation* dengan membagi sub AS seperti pada Gambar 3 dibawah.



Gambar 3. Topologi menggunakan metode *confederation*

2.2.2. Metode Route reflector

Sama halnya dengan *confederation route reflector* merupakan metode lain untuk menangani masalah full mesh [5]. Menggunakan *router reflector* ini mengurangi jumlah peer BGP dan pesan BGP dalam satu reflektor rute AS dan bekerja persis sama dengan Designated *Router* (DR) di OSPF, jadi ada 1 *router* yang merupakan pusat perutean, jadi *router* BGP speaker adalah peer ke semua *router* BGP yang ada Hanya reflektor peer-to-route yang tidak memerlukan koneksi peer-to-peer yang memungkinkan. Reflektor rute mencerminkan rute yang dianggap sebagai yang terbaik saja. Selain itu, reflektor rute tidak diperbolehkan untuk mengubah atribut rute yang direfleksikan, termasuk atribut next hop. Contoh pada Gambar 2 juga dapat di selesaikan dengan metode *route reflector* seperti Gambar 4 dimana R4 menjadi *router* reflectornya.



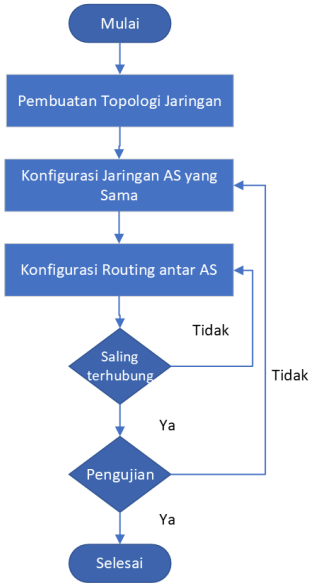
Gambar 4. Topologi menggunakan metode *Route reflector*

2.2.3. Metode Next hop self

Next Hop Self adalah salah satu konfigurasi pada routing BGP yang sangat penting. Tujuannya adalah agar gateway dari sebuah router untuk melewati router yang mengaktifkan nexthop self ini akan digunakan sebagai gateway. Pada topologi gambar 4 router R4 akan mengaktifkan next hop self kepada router R6, R7, dan R5 agar ketiga router itu dapat menggunakan router R4 sebagai gateway.

2.3. Perancangan Penerapan Metode

Langkah selanjutnya adalah pembuatan flowchart yang akan disajikan pada gambar 5.



Gambar 5. Flowchart perancangan dan implementasi.

Pada Gambar 5 dijelaskan bahwa tahap pertama adalah pembuatan topologi jaringan menggunakan metode Metode *Confederation* , Metode *Route reflector* , dan Metode *Next hop self* pada tiap tiap AS. Setelah dibuat desain topologinya kemudian akan dikonfigurasi masing-masing AS agar dapat terhubung satu sama lain. Setelah sudah terhubung, Langkah selanjutnya adalah mengkonfigurasi antar AS yang berbeda agar dapat terhubung satu sama lain. Setelah semua terhubung langkah terakhir adalah melakukan pengujian sistem menggunakan command prompt di komputer untuk menguji apakah setiap perangkat dapat berkomunikasi satu sama lain dan meminimalisir *routing* tabel pada *router*. Jika hasilnya sudah sesuai dengan apa yang diinginkan maka proses implementasi tersebut telah selesai dan dapat digunakan.

2.4. Quality of Service

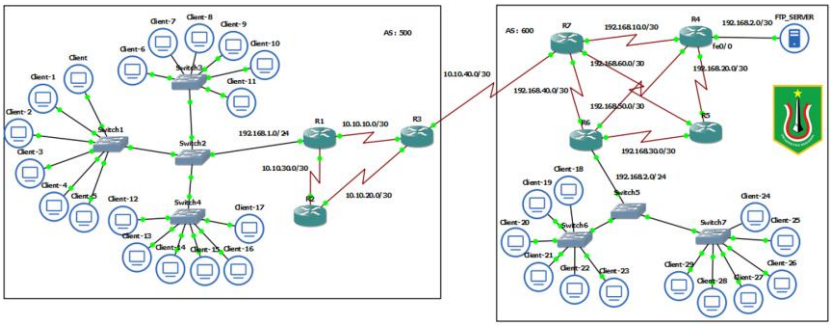
QoS (Quality of Service) didefinisikan sebagai sebuah mekanisme yang memungkinkan layanan dapat beroperasi sesuai dengan karakteristiknya [9]. Untuk melakukan analisis QoS pada jaringan, penulis menggunakan aplikasi wireshark. Wireshark adalah packet analyzer open source, Tools ini seringkali digunakan untuk menemukan masalah pada jaringan, pengembangan perangkat lunak dan protokol komunikasi [10]. Penulis melakukan beberapa parameter diantaranya [11] :

a. Waktu yang dibutuhkan untuk sebuah paket data diteruskan ke hop berikutnya (*forwarding time*)
b. Waktu tunda suatu packet data yang diakibatkan oleh proses transmisi dari satu tempat ke tempat lainnya (*delay*)
c. Jumlah bit data yang diterima dalam selang waktu tertentu dengan satuan byte per second (*throughput*)

3. HASIL DAN PEMBAHASAN

3.1. Analisa

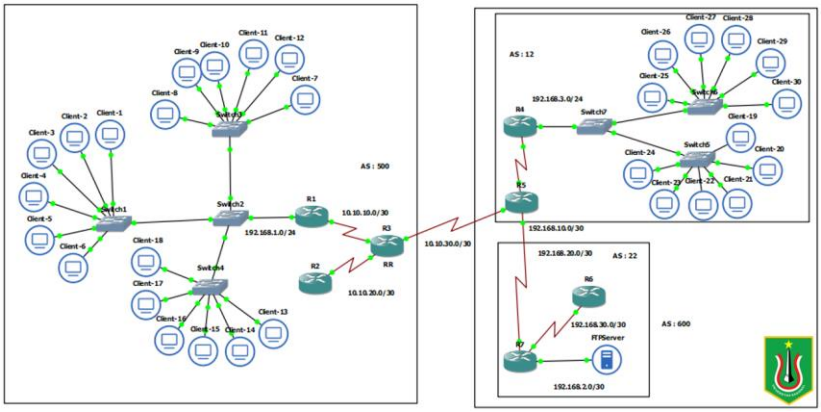
Penulis sudah menganalisa berdasarkan beberapa literatur yang relevan bahwa dalam iBGP menjadi sangat wajib dikarenakan iBGP memiliki aturan split horizon dimana saat *router* menerima update dari *router* iBGP lain, maka *router* tersebut tidak akan memforward paket tersebut ke *router* iBGP lain. Dalam hal ini akan disimulasikan topologi jaringan full mesh seperti Gbr.6 di bawah ini :



Gambar 6. Topologi Jaringan *Fullmesh*

3.2. Desain Topologi

Berikut ini akan diterapkan metode *confederation*, metode *route reflector* dan metode *next hop self* pada Border Gateway Protocol (BGP) berdasarkan topologi full mesh pada Gbr.6 menjadi seperti Gbr.7 di bawah ini :



Gambar 7. Topologi Jaringan Metode *Confederation* , *Route reflector* dan *Next hop self*

Pada topologi pada gambar 7 di atas memiliki 7 buah *router* 3 buah *router* pada AS 500 dan 4 buah router pada AS 600 , 1 buah komputer client dan 1 buah server. Setiap *router* memiliki alamat IP unik pada port yang diaktifkan. Server adalah sesuatu yang akan diakses oleh komputer klien. Protokol *routing* yang digunakan digunakan pada AS 500 adalah BGP yang menggunakan metode *Route reflector* dan *Next hop self* dimana menjadikan *router* R3 menjadi *Router* Reflector. Sedangkan pada AS 600 adalah BGP yang menggunakan metode *confederation* dan *Next hop self* dimana membagi AS 600 menjadi sub AS 12 dan Sub As 22.

3.3. Implementasi

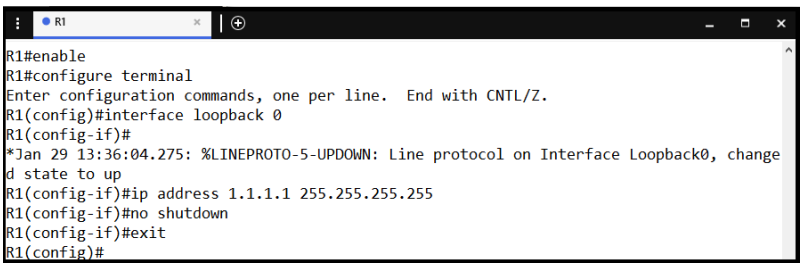
Konfigurasi IP Address

Setelah membuat desain topologi selanjutnya adalah konfigurasi *router*. Tabel 1 berikut ini adalah konfigurasi IP address pada masing-masing perangkat.

Tabel I
Konfigurasi IP Address Perangkat

No	Nama Device	Port	IP Address
1	*Router* R1	Se 1/0	10.10.10.1/30
		Fe 0/0	192.168.1.1
		Loopback 0	1.1.1.1/32
2	*Router* R2	Se 1/0	10.10.20.1/30
		Loopback 0	2.2.2.2/32
3	*Router* R3	Se 1/0	10.10.10.2/30
		Se 1/1	10.10.20.2/30
		Se 1/2	10.10.30.1/30
		Loopback 0	3.3.3.3/32
4	*Router* R4	Se 1/0	192.168.10.2/30
		Loopback 0	4.4.4.4/32
5	*Router* R5	Se 1/0	192.168.10.1/30
		Se 1/1	192.168.20.1/30
		Se 1/2	10.10.30.1/30
		Loopback 0	5.5.5.5/32
6	*Router* R6	Se 1/0	192.168.30.1/30
		Loopback 0	6.6.6.6/32
7	*Router* R7	Se 1/0	192.168.20.2/30
		Se 1/1	192.168.30.2/30
		Fe 0/0	192.168.2.1
		Loopback 0	7.7.7.7/32
8	Client	Ethernet 0	192.168.1.2
9	Server	Ethernet 0	192.168.2.2

Mengacu pada tabel di atas, terdapat istilah "IP loopback". Alamat IP loopback adalah alamat virtual (semu) yang dapat digunakan untuk mengidentifikasi *router* di jaringan. Dengan kata lain, antarmuka loopback sebenarnya tidak ada karena merupakan antarmuka logis dan bukan antarmuka fisik. Loopback tidak memiliki kabel fisik yang terhubung ke *router* atau switch. Loopback adalah antarmuka yang tidak dalam posisi "down", tetapi dapat menonaktifkan atau menonaktifkan antarmuka fisik jika terjadi kesalahan koneksi. Oleh karena itu, alamat IP antarmuka loopback sangat cocok untuk digunakan sebagai ID *router*. Alamat IP loopback diatur ke alamat IP tertinggi dengan subnet mask 255.255.255.255. Berikut ini adalah konfigurasi pada interface loopback pada *router* dan dapat diterapkan di interface lainnya.

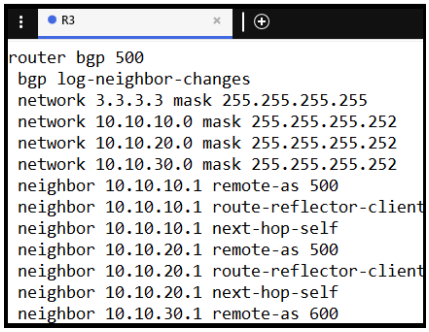


Gambar 8. Konfigurasi Loopback *router* R1

Konfigurasi Router BGP

Pada tahap ini akan membahas bagaimana konfigurasi terhadap *router* R3, R5 dan R7 menggunakan protokol *routing* BGP dengan metode *confederation, route reflector, next hop self*. *Router* R5 dan R7 akan menjadi sub AS pada metode *confederation* dalam AS 600, sedangkan *Router* R3 ditetapkan menjadi *router* reflector dalam AS 500. Konfigurasinya adalah sebagai berikut :

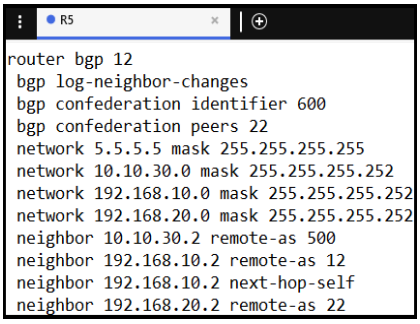
a. Konfigurasi *routing* BGP pada *router* R3 :



Gambar 9. Konfigurasi *routing* bgp R1

Dalam menggunakan *route reflector* (mirror) memerlukan client/mirror untuk ditangkap. Pada gambar di atas menetapkan *neighbor router* R1(10.10.10.1) dan *router* R2(10.10.20.1) untuk saling menangkap melalui kata kunci route-reflector-client dan kedua *neighbor* tersebut menggunakan *next-hop-self* agar kedua *neighbor* itu menggunakan *router* R3 sebagai gateway.

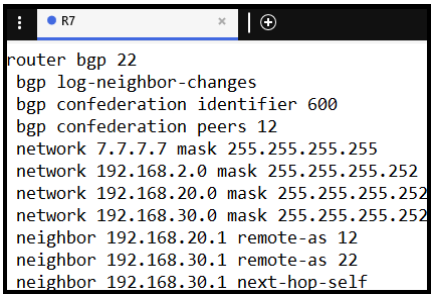
b. Konfigurasi *routing* BGP pada *router* R5 :



Gambar 10. Konfigurasi *routing* bgp R2

Pada gambar 10 di atas terdapat *routing confederation* identifier 600 hal itu bermaksud untuk memberi tahu *router* ini bahwa milik AS 600 (AS Utama). Sedangkan *confederation* peers 22 berarti sub-AS tersebut sebagai *neighbor* sub-AS yang terhubung langsung. *Router* R5 juga memberikan izin terhadap *neighbor router* R6 (192.168.10.2) untuk menjadikan *router* R5 sebagai gateway dengan command *next-hop-self*.

c. Konfigurasi *routing* BGP pada *router* R7 :



Gambar 11. Konfigurasi *routing* bgp R3

Sama halnya dengan *router* R5 pada *router* R7 berlaku demikian. Pada gambar 11 juga terdapat *confederation* peers 12 yang berarti sub-AS 12 menjadi *neighbor* sub-AS 22 yang terhubung langsung. Pada konfigurasi *router* R7 juga menggunakan command next-hop-self pada *router* R6 (192.168.30.1) agar *router* R6 dapat menjadikan *router* R7 sebagai *gateway*.

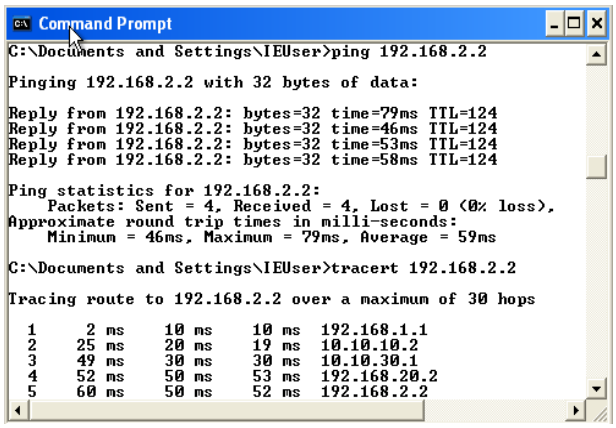
3.4. Pengujian Sistem Konfigurasi

Setelah melakukan konfigurasi maka langkah selanjutnya adalah melakukan testing atau pengujian apakah jaringan tersebut terhubung satu sama lain serta menghitung forwarding delay, delay , dan throughput.

Menguji Koneksi

Pada pengujian ini, akan dilakukan tiga pengujian. Pengujian client ke server , pengujian *routing* reflector pada AS 500 , dan pengujian *confederation* pada AS 600 menggunakan tools PING dan TRACEROUTE.

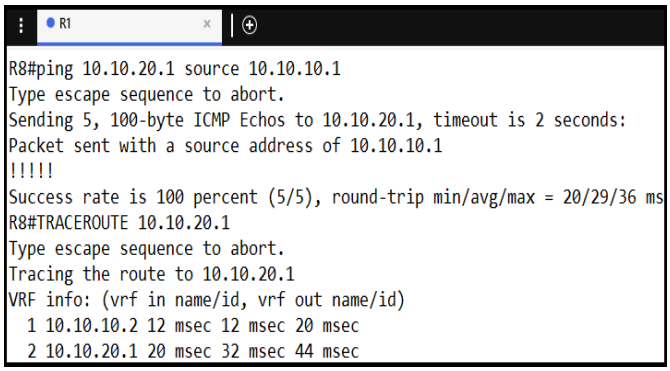
Pengujian pertama adalah melakukan Ping pada komputer Clien ke komputer Server.



Gambar 12**.** Ping pada Client ke Server

Pada Gambar 12 dapat diambil kesimpulan bahwa koneksi antar clien dengan server sudah terhubung bisa saling membalas ping yang dikirimkan dari masing-masing clien maupun server.

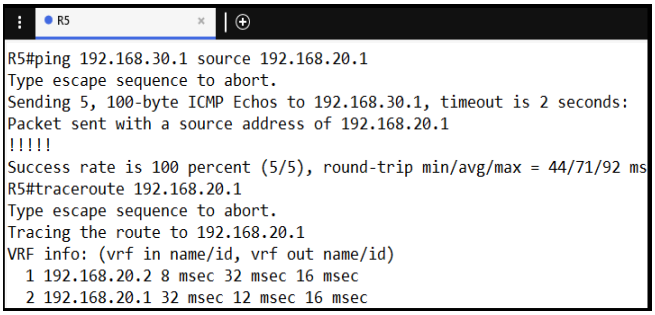
Pengujian kedua adalah melakukan Ping pada setiap *router* AS 500.



Gambar 13. Ping pada *router* R1 ke *router* R2

Pada Gambar 13 dapat diambil kesimpulan bahwa koneksi antar *router* R1 dengan *router* R2 sudah terhubung bisa saling membalas ping yang dikirimkan dari masing-masing clien maupun server dan berarti konfigurasi *route reflector* dan next hop pada AS ini berhasil.

Pengujian ketiga adalah melakukan Ping pada setiap *router* AS 600.



Gambar 14. Ping pada *router* R5 ke *router* R6

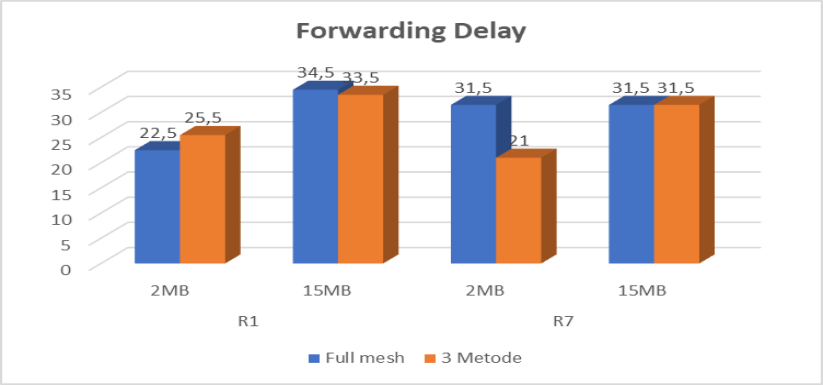
Pada Gambar 14 dapat diambil kesimpulan bahwa koneksi antar *router* R5 dengan *router* R6 sudah terhubung bisa saling membalas ping yang dikirimkan dari masing-masing clien maupun server dan berarti konfigurasi route *confederation* dan next hop pada AS ini berhasil.

Pengukuran Analisis Quality of Service

a. Forwarding Delay

Penulis melakukan pengujian dengan mengirimkan data antara client ke server sebesar 2Mb dan 15 Mb. Proses pengambilan data di ambil di interface inbound dan outbound pada *router* yang dilalui packet data yaitu *router* R1 dan R7. Selanjutnya dilakukan perhitungan selisih waktu antara packet data masuk dan packet data keluar dan hasilnya yang dijadikan sebagai acuan *forwarding delay.*

Berikut ini adalah grafik hasil pengujian untuk forwarding delay pada BGP *fullmesh* dan BGP dengan ketiga metode :



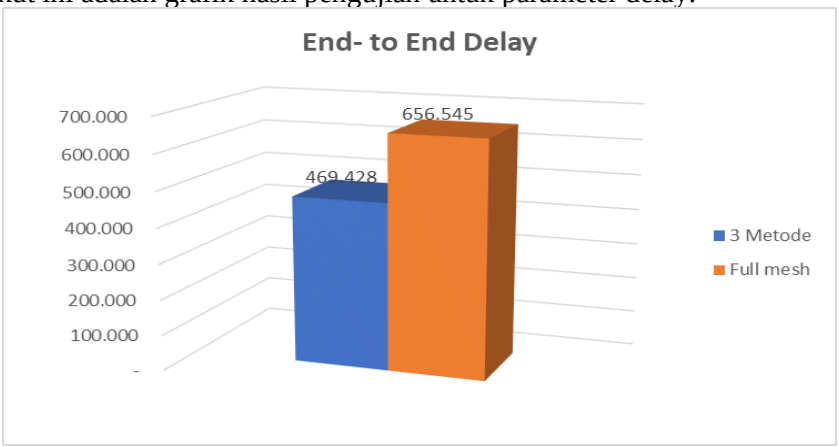
Gambar 15. Grafik Forwarding Delay

Pada Gambar 15 dapat disimpulkan bahwa dalam forwarding delay BGP yang menggunakan metode lebih rendah 7% dibandingkan dengan BGP *fullmesh*. Hal ini dikarenakan besarnya tabel *routing* berpengaruh pada kecepatan proses forwarding itu sendiri. Alasannya semakin banyak sesi, semakin banyak rute yang ada pada tabel forwarding sehingga hal ini mempengaruhi proses lookup tabel forwarding.

b. End to End Delay

Delay yang digunakan pada penelitian ini adalah end to end delay yaitu waktu yang dibutuhkan oleh paket data untuk mencapai destinasi. Pengukuran delay dilakukan dengan mengirimkan paket ICMP antara client ke server dengan menggunakan perintah ping kemudian dilakukan perhitungan terhadap waktu yang dibutuhkan saat melewati jaringan.

Berikut ini adalah grafik hasil pengujian untuk parameter delay.



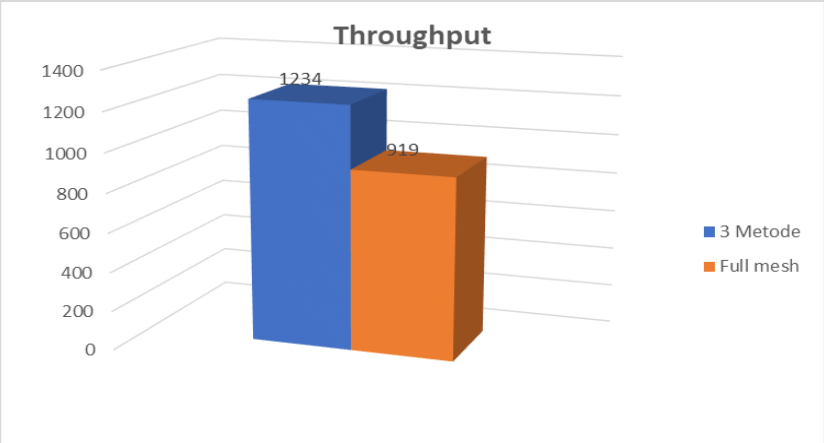
Gambar 16. Grafik Forwarding Delay

Jumlah delay yang dihasilkan dari pengiriman packet data dipengaruhi beberapa hal salah satunya adalah tergantung jumlah hop yang harus dilalui oleh packet data. Dari gambar 16 dapat dilihat BGP menggunakan 3 metode memiliki nilai lebih baik dibandingkan BGP *fullmesh*. Pada kasus ini jumlah *router* yang dilewati oleh kedua BGP sama yaitu melewati empat buah *router*. Hal ini disebabkan lebih cepatnya proses forwarding packet data. Maka dapat disimpulkan bahwa pada pengukuran delay BGP menggunakan tiga metode lebih rendah 29% dibandingkan BGP *fullmesh*.

c. Throughput

Pengujian ini dilakukan bersamaan saat melakukan end-to end delay dengan mengirim packet sebanyak 100 antara client dan server. Tujuan dari pengukuran ini adalah untuk melihat kehandalan jaringan pada saat meneruskan paket dari sumber ke tujuan.

Berikut ini adalah hasil pengujian throughput pada kedua BGP :



Gambar 17. Grafik Throughput

Besarnya nilai throughput berbanding terbalik dengan delay, artinya semakin cepat delay yang dihasilkan maka semakin besar nilai throughput yang dihasilkan. Dari gambar di atas dapat dilihat bahwa nilai throughput BGP yang menggunakan 3 metode lebih besar dibandingkan BGP *fullmesh*. Hal ini dikarenakan lebih sedikitnya tabel forwarding yang harus di lookup, maka proses lookup informasi *routing* menjadi lebih singkat dan proses pengiriman packet data menjadi jauh lebih cepat. Terbukti pada penelitian ini throughput BGP menggunakan tiga metode lebih tinggi 26% dibandingkan BGP *fullmesh*.

4. KESIMPULAN

Berdasarkan simulasi yang telah diterapkan dan dilakukan perbandingan antara BGP menggunakan ketiga metode dengan BGP *fullmesh* dapat disimpulkan sebagai berikut :

a. Dari hasil penerapan simulasi terbukti bahwa ketiga metode ini dapat digunakan dan menghubungkan AS yang sama maupun AS yang berbeda.
b. Hasil pengukuran forwarding delay menunjukkan BGP menggunakan ketiga metode 7% lebih rendah dibandingkan BGP *fullmesh*. Pada pengujian end to end delay pun lebih rendah 29% dibandingkan full mesh. Dan pada pengukuran throughput terbukti lebih tinggi 26%. Hal ini membuktikan ketiga metode ini dapat mengurangi tabel *routing* dan membuat pengiriman packet data menjadi lebih cepat.
c. Hasil pengujian pada poin dua menunjukan bahwa penerapan ketiga metode ini dapat menjadi salah satu solusi dalam menyelesaikan permasalahan *fullmesh* pada iBGP

5. SARAN

Berdasarkan hasil yang sudah didapatkan dari penelitian ini, beberapa rekomendasi disarankan oleh penulis sebagai berikut:

a. Penelitian selanjutnya diharapkan diterapkan pada studi kasus nyata.
b. Dapat dilakukan penelitian dengan topologi yang lebih kompleks , hop yang dilewati lebih banyak untuk dilintasi.
c. Dapat menambahkan metode lain pada *routing* BGP untuk memecahkan permasalahan *fullmesh*.
d. Dapat membandingkan BGP menggunakan ketiga metode ini dengan *routing* protocol lainnya.

DAFTAR PUSTAKA

[1] E. R. Saputra, “Analisis Performansi Jaringan Autonomous System Dengan Metode Confederations (RFC 5065),” 2011.
[2] Wikipedia, “Border Gateway Protocol,” *en.wikipedia.org*, 2021. https://en.wikipedia.org/wiki/Border_Gateway_Protocol#cite_note-17.
[3] T. Rekhter, Y., & Li, “A Border Gateway Protocol 4 (BGP-4). RFC 1771,” 1995.
[4] J. Traina, P., McPherson, D., & Scudder, “Autonomous System Confederations for BGP. RFC 5065,” 2007.
[5] R. Bates, T., Chen, E., & Chandra, “BGP Route Reflection: An Alternative to Fullmesh Internal BGP (IBGP). RFC 4456.,” 2006.
[6] H. A. Musril, “Simulasi Interkoneksi Antara Autonomous System (As) Menggunakan Border Gateway Protocol (Bgp),” *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 2, no. 1, pp. 1–9, 2017, doi: 10.30743/infotekjar.v2i1.151.
[7] D. Darmawan and T. Imanto, “Analisa Link Balancing dan Failover 2 Provider Menggunakan Border Gateway Protocol (BGP) Pada Router Cisco 7606s,” *J. Nas. Teknol. dan Sist. Inf.*, vol. 3, no. 3, pp. 326–333, 2017, doi: 10.25077/teknosi.v3i3.2017.326-333.
[8] U. Black, *IP Routing Protocols RIP, OSPF, BGP, PNNI & CISCO ROUTING PROTOCOL*. *New Jersey*. New Jersey, 2000.
[9] T. Ernawati and J. Endrawan, “Peningkatan Kinerja Jaringan Komputer dengan Border Gateway Protocol (BGP) dan Dynamic Routing (Studi Kasus PT Estiko Ramanda),” *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 4, no. 1, p. 35, 2018, doi: 10.23917/khif.v4i1.5656.
[10] E. W. Richard Sharpe, “Wireshark User ’ s Guide,” p. 191, 2014.
[11] R. Wulandari, “Analisis QoS (Quality of Service) pada Jaringan Internet UPT Loka Uji Teknik Penambangan-LIPI),” *J. Tek. Inform. dan Sist. Inf.*, vol. 2, no. 2, pp. 162–172, 2016.