

Perbandingan Redistribusi *Routing* Protokol Dinamis pada Exterior Gateway Protokol

Dadang Wahyudi¹, Dedy Syamsuar², Edi Surya Negara³

Program Magister Teknik Informatika
Universitas Bina Darma

email: ¹Dadangwahyudi57@gmail.com, ³edisuryanegararahap@gmail.com

Jl. A. Yani No. 12, Palembang 30624,

Abstrak

Setiap protokol routing memiliki algoritma dan metrik yang berbeda dalam menentukan rute pada jaringan komputer tersebut. Dengan menggunakan teknologi redistribusi pada protokol routing maka akan terbentuk sebuah infrastruktur jaringan komputer yang menggunakan routing berbeda pada jaringan komputer itu. Dengan adanya redistribusi maka setiap paket yang sedang beraktifitas pada masing-masing routing akan berpengaruh pada pemakaian bandwidth, broadcast, tingkat stabilitas, dan paket-paket tersebut. Pada penelitian ini penulis akan merancang sebuah jaringan komputer dan membandingkan protokol *routing internal* yang bekerja melewati rute *external* dapat ukur dengan melihat tingkat stabilitas, paket dikirim dan diterima, pemakaian *bandwidth*.

Kata kunci : Routing Protokol, Redistribusi, EIGRP, OSPF, EGP

1. PENDAHULUAN

Protokol *routing* dalam jaringan komputer menjadi salah satu variabel dalam menentukan kualitas kinerja jaringan. Semakin cepat suatu protokol routing dalam menyampaikan informasi melalui *route* yang ada dalam tabel *routing* (konvergensi), maka akan semakin baik kualitas pada jaringan tersebut. Secara umum ada dua katagori *table routing protocol* yaitu *routing* statis dan *routing* dinamis. *Routing* statik adalah *routing* yang memerlukan campur tangan administrator jaringan dan *routing* tersebut bekerja secara manual dalam melakukan proses dan pertukaran informasi dengan *router* tetangganya. Sedangkan *routing* dinamis, protokol routing ini dapat menentukan sendiri *route* berdasarkan situasi dan kondisi setiap saat. *Routing* dinamis yang digunakan oleh *network* kemudian dibentuk oleh beberapa buah *router*, dan masing-masing *router* akan saling memberi informasi pada *router* tetangganya kemudian *router* tersebut bersama-sama membentuk *routing* tabel. Pada *routing* dinamis tersebut terdapat dua jenis *routing* protokol yaitu *link state* dan *distance vector* (Sofana, 2009).

Pada umumnya *protocol routing* mendistribusikan tabel *routing*-nya sendiri ke *router* yang lain. Untuk menyingkronkan *table routing*, maka beberapa *routing* di izinkan ke *router-router* yang lain untuk meng-*update* secara periodik tentang status jaringan mereka. Gabungan antara algoritma *static* dan *dynamic* dapat meningkatkan *performance* dari jaringan tersebut.

Algoritma *routing static* merupakan suatu algoritma yang diatur oleh administrator jaringan tersebut untuk mengizinkan melakukan *routing* paket ke jaringan melalui *router* tertentu. Algoritma ini tidak bisa memilih jalan yang optimal, *routing static* biasanya digunakan untuk jaringan yang kemungkinan kecil mengalami perubahan dalam topologinya. Sedangkan algoritma *dynamic* dapat diklasifikasikan menjadi satu dari dua kategori *Distance Vector* dan *Link State*. *Routing Distance Vector* merupakan jenis protokol *routing* lama. Beberapa ciri khas *distance vector* :

- 1) *Distance* Atau jarak mencapai tujuan akhir. *Distance* dapat ditemukan berdasarkan *cost* yang ditentukan dari jumlah *host* (hitungan *hop*). Yang dilalui rute atau jumlah total perhitungan *metric* pada rute tersebut. Informasi diperoleh dari *router* tetangga yang terhubung langsung dengannya.
- 2) *Vector* merupakan arah *traffic*. Ketika data akan di-*forward* ketujuan maka data tersebut pasti akan melalui *network interface* hingga dapat mencapai tujuan
- 3) Perubahan topologi *network* biasanya akan direspon oleh *protocol* secara *slow converge*.
- 4) Tidak mendukung *Variable Length Subnet Mask* (VLSM) dan *Class inter Domain Routing* (CIDR).
- 5) Tidak mudah di implementasi pada jaringan berskala besar.
- 6) Mendukung Algoritma *Bellman* dan *Ford*.

Sedangkan pada *routing protocol Link-state* merupakan jenis protokol yang lebih baru, beberapa ciri protokol *Link-state* :

- 1) Protokol ini dapat menentukan status dan tipe koneksi setiap *link* dan menghasilkan sebuah perhitungan *metric*,
- 2) Protokol dapat mengetahui apakah *link* sedang *up* atau *down*, dan dapat mengetahui seberapa cepat untuk mencapai tujuannya.
- 3) Dapat mengetahui perubahan topologi jaringan dengan cepat, disebut *fast Converge*.
- 4) mendukung *Variable Length Subnet Mask* (VLSM) dan *Class inter Domain Routing* (CIDR).
- 5) Dapat di implementasi dalam jaringan berskala besar.
- 6) Menggunakan algoritma *Dijkstra*.

Adapun Fitur-fitur yang dimiliki *routing protocol link-state* itu sendiri adalah sebagai berikut :

- 1) *Link-state Advertisement* (LSA) adalah paket kecil dari informasi *routing* yang dikirim antar-*router*,
- 2) Topologi *database* adalah kumpulan informasi dari LSA,
- 3) *SPF Algorithm* adalah hasil perhitungan pada *database* sebagai hasil dari pohon SPF,
- 4) *Routing table* adalah daftar rute dan *interface*

Protocol routing seperti RIP, IGRP, EIGRP, dan OSPF merupakan algoritma *routing* yang *dynamic*. Dalam protocol tersebut secara periodik meng-*update* dan menganalisis dengan cara menerima paket dari router lain jika terjadi perubahan dalam topologi suatu jaringan (Yugianto, 2012).

Routing protokol *dynamic* itu sendiri terdapat dua algoritma yaitu algoritma bellman-ford dan algoritma Dijkstra.. Algoritma *Bellman-Ford* yang digunakan secara terdistribusi adalah berdasar pada ide bahwa jika suatu node *router* B terdapat dalam jalur terdekat antara node A dan C, maka jalur dari node A ke node B pasti juga merupakan jalur terdekat. Demikian pula jalur dari node C ke node B. Penggunaan algoritmanya adalah sebagai berikut:

Didefinisikan:

$dx(y)$ = jarak terdekat antara node x ke node y.

$c(x,y)$ = jarak antara node x ke node y.

Inisialisasi:

$dx(v) = \infty$, untuk semua $v \neq x$

$dx(x) = 0$

Update nilai:

Untuk dapat mencapai node z dari x $d_x(z) = \min \{c(x,v) + dv(z)\}$ Dimana v merupakan semua node tetangga dari node x.

Contoh: Pada jaringan contoh_1 dicari jalur terdekat dari node router x ke node z.

Kita ketahui bahwa $du(z) = 3$ dan $dv(z) = 2$

Jalur terdekat dari x ke z:

$dx(z) = \min \{c(x,u) + du(z) , c(x,v) + dv(z)\}$

$= \min \{1 + 3 , 3 + 2\}$

$= \min \{4 , 5\}$

Maka didapatlah bobot jalur terdekat dari node x ke z adalah 4 dan *router* berikutnya yang harus diambil dari node x adalah node u. Dari contoh di atas terlihat bahwa node x memelihara bobot jalur minimum ke node z yang di dalamnya sekaligus memelihara bobot jalur minimum dari node u dan v (yang merupakan tetangga dari node x) ke node z. Dalam hal ini *algoritma Bellman-Ford* yang digunakan didefinisikan secara rekursif. (Kristianto, 2012)

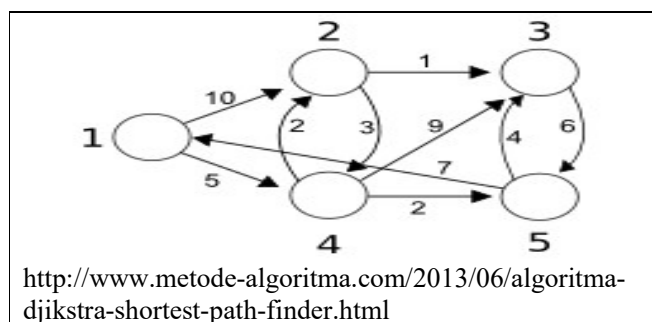
Algoritma bellman-ford menghitung jarak terpendek “dari satu sumber” maksudnya dari satu sumber ialah bahwa ia menghitung semua jarak terpendek yang berawal dari satu titik node. Sedangkan

algoritma *Dijkstra* dapat lebih cepat mencari hal yang sama dengan syarat tidak ada sisi “*edge*” yang berbobot negatif. Maka algoritma *Bellman-Ford* menggunakan waktu sebesar $O(V.E)$, dimana V dan E adalah banyaknya sisi dan titik, dalam konteks ini bobot ekuivalen dengan jarak dalam sebuah sisi.

Algoritma *dijkstra* pertama kali dikembangkan oleh *E.W. Dijkstra*. Pada perkembangannya, algoritma ini menggunakan struktur data yang berbeda-beda. Pada umumnya, graph setidaknya mempunyai satu arc penghubung dari satu node ke node yang lain.

Secara garis besar algoritma *dijkstra* membagi semua node menjadi dua, kemudian dimasukkan ke dalam tabel yang berbeda, yaitu tabel permanen dan tabel temporal. Tabel permanen berisi node awal dan node-node yang telah melalui proses pemeriksaan dan labelnya telah diubah dari temporal menjadi permanen. Tabel temporal berisi node-node yang berhubungan dengan node pada tabel permanen. Pemilihan rute algoritma *dijkstra* dilakukan dengan *Best First Search (BFS)*, dimana sebuah rute akan dihitung jaraknya dari node awal ke node lain, dalam suatu jaringan, kemudian rute-rute ini akan dibandingkan, dan rute dengan jarak yang paling pendek akan dipilih sebagai rute terpendek. Proses ini akan terus berlangsung secara iterasi dan akan berhenti ketika mencapai node tujuan. Pada algoritma *dijkstra*, node-node dibagi menjadi dua set. Set pertama berisi node yang telah diperiksa, sedangkan set kedua berisi node yang belum diperiksa. Untuk menyimpan node-node itu dapat digunakan struktur data *array* dan *priority queue* dengan elemen yang mempunyai urutan tertentu, yaitu *ascending* atau *descending* (Purwananto dkk, 2005)

Algoritma *dijkstra* atau algoritma *Shortest Path First (SPF)*. Algoritma ini memperbaiki informasi *database* dari informasi topologi. Algoritma *Dijkstra* dapat menemukan jalur terpendek untuk mencapai kesemua tujuan akan tetapi algoritma ini hanya mencari jalur terpendek tanpa melihat jalur tersebut sedang sibuk atau dalam keadaan yang tidak baik. Contoh Algoritma *Dijkstra* dapat dilihat pada gambar di bawah ini :



Gambar 1: Algoritma Dijkstra

Pada gambar di atas dapat dilihat setiap node memiliki beberapa rute untuk menentukan rute terdekat, contoh gambar diatas dengan merekayasa node 1 menuju node 3. Ada beberapa jalur (1 -> 4-> 3, 1-> 2-> 3) tetapi jalur terpendek dari mereka adalah 1 -> 4-> 2-> 3 dari panjang 9.

Secara konsep biasanya pada satu infrastruktur jaringan terdapat satu *routing* protokol yang bisa digunakan. Akan tetapi dengan menggunakan metode *redistribution routing protocol* pada satu jaringan, maka jaringan tersebut bisa menggunakan *routing* yang berbeda-beda. Pada saat menerapkan teknologi redistribusi di-jaringan, dan disaat *router* meneruskan paket ke *router* tetangganya, maka pada tingkatan *stability*, *bandwidth*, *packet*, dan *broadcast* pada masing-masing *routing protocol* akan berbeda pula.

Pertukaran informasi dengan menggunakan konsep *route redistributions* pada *dynamic routing protocol* memungkinkan kita untuk memprediksi *route* mana yang dipilih oleh paket data tersebut. Pemahaman ini dapat mengungkapkan pembentukan *osilasi*, *routing loop* dan ketidakstabilan lainnya. Model ini berlaku untuk semua *routing* protokol yang ada, terkecuali *BGP*. Karena *redistribution routing protocol* dalam hal menentukan *route* dan proses *BGP*, pertama terlihat pada *BGP AS_PATH*. Untuk memutuskan apakah anggapan *routing protocol* lain dapat dipertimbangkan disemua *router* dan paket data yang masuk dapat diteruskan ke *router* selanjutnya. *Protocol routing BGP* juga berbeda dari *protocol routing* yang lain masing-masing vendor memiliki definisi, dan dapat menetapkan sendiri pilihan untuk mendistribusikan *routing BGP* ke *protocol IGP* yang ada. Beberapa *network* yang dibangun menggunakan protokol *routing* yang berbeda dan dapat saling dihubungkan menggunakan teknik tertentu yang disebut *route redistribution*, artinya kita akan mendistribusikan ulang rute sesuai kondisi tertentu (Sofana, 2012).

Redistribution adalah penggunaan protokol *routing* untuk mengiklankan *route* yang dipelajari dengan cara lain, misalnya dengan *routing protocol* lain, *route* statis, atau terhubung langsung *route*. Dalam beberapa kasus, membuat redistribusi menjadi sebuah keharusan. Perbedaan karakteristik *routing protocol*,

seperti metrik, jarak administratif, *classful* dan *classless* dapat mempengaruhi redistribusi. Pertimbangan harus diberikan untuk perbedaan ini agar redistribusi berhasil (*Cisco System, 2014-2015*)

Beberapa alasan yang menyebabkan kita harus melibatkan berbagai protokol *routing* antara lain :

- 1) Menggunakan aplikasi yang hanya berjalan pada protokol *routing* tertentu,
- 2) Menggunakan *hardware* dari berbagai *vendor* yang harus mengharuskan penggabungan protokol,
- 3) Harus menghubungkan *network* dengan area atau *domain routing* yang dikelola oleh orang lain,
- 4) Melakukan proses migrasi yang melibatkan beberapa protokol *routing*.

Cukup banyak teknik yang disediakan dan masing-masing cukup kompleks, dan bagai mana cara mendistribusikan *traffic* yang dapat melibatkan dua atau lebih *routing domain* berbeda. Ada beberapa cara atau teknik untuk mengendalikan atau *menentukan path/ rute*, seperti: *Route maps, Prefix List, Distribute List, Passive Interface, Administrative Distance, Offset-list, IOS IP SLA, Policy Based Routing*.

Pada penelitian ini penulis akan melakukan perbandingan redistribusi protokol *routing* dinamis pada *Exterior Gateway Protocol*, dengan demikian yang menjadi tolak ukur dalam membandingkan *protocol routing internal* yang bekerja melewati *route external* dapat ukur dengan melihat tingkat *stability, packet* yang dikirim dan paket yang diterima, pemakaian *bandwidth*,. Adapun judul yang diangkat pada penelitian ini adalah “Perbandingan Redistribusi *Routing* Protokol Dinamis pada *Exterior Gateway Protocol*”.

2. METODOLOGI PENELITIAN

2.1 Metode Penelitian

Metode penelitian ini menggunakan metode *Experiment research* yaitu mengadakan percobaan untuk melihat hasil suatu hasil. Adapun langkah-langkah dalam penelitian *Experiment research* pada dasarnya hampir sama dengan penelitian lainnya, dalam penelitian *experiment* yang perlu ditekankan adalah sebagai berikut :

- 1) Adanya permasalahan yang signifikan untuk diteliti
- 2) Pembuatan atau pengembangan instrumen
- 3) Pemilihan desain penelitian
- 4) Melakukan analisis data
- 5) Memformulasikan simpulan

2.2 Metode Pengumpulan Data

Adapun tahapan-tahapan yang digunakan untuk melakukan pengumpulan data adalah sebagai berikut:

- 1) *Study literature*. Dengan memperoleh melalui studi keperpustakaan dalam mencari bahan bacaan dari buku, jurnal dan internet yang berkaitan dengan penelitian serta parameter yang akan diteliti
- 2) Pengamatan. Pengumpulan data dengan cara observasi mengirim paket data ping host ke server yang melakukan trafik jaringan
- 3) Uji coba. Data-data kinerja router redistribusi pada protokol dinamik didapatkan dari semua aktifitas semua paket data ping

2.3 Metode Pengukuran

Saat melakukan penelitian di jaringan komputer yang telah dirancang peneliti menggunakan *routing EIGRP* pada area *Distance Vector*, *routing OSPF* dibagian *Link-state* sedangkan untuk *exterior* Peneliti menggunakan *routing BGP*. Agar peneliti mengetahui kualitas pada setiap masing-masing *routing* peneliti menggunakan software *axenettols* dan menetapkan metode pengukuran sebagai berikut :

- 1) Menghubungkan *interfaces* antara *router, server* dan *client*
- 2) Menetapkan Jumlah paket yang akan didownload dari Server FTP adalah sebesar 4.90GB atau sama dengan 5.140.922kb.
- 3) Masing-masing *bandwidth* yang telah ditetapkan disetiap interface sebesar 10mb atau sama dengan 10000kb
- 4) Pengukuran paket ditentukan dengan waktu selama 10 menit dalam keadaan jaringan sedang sibuk.
- 5) Saat Melakukan Pengukuran peneliti menggunakan 1 komputer Client yang sedang melakukan download data dari FTP Server masing-masing client mendownload paket sebesar 4.90gb atau sama dengan 5.140.922kb.
- 6) Menggunakan software *Axenettols Profesional* versi 4.0.10.
- 7) Mengambil trafik dari *client* ke *server* disaat jaringan sibuk/ sedang bekerja.

- 8) Membandingkan setiap Paket, Bandwidth, dan melihat stabilitas jaringan, dari setiap routing yang diteliti.
- 9) Dokumentasi.

Dari penjelasan diatas, software axenettols adalah sebagai media monitoring kualitas jaringan pada setiap paket yang sedang berjalan melewati masing-masing protokol routing yang berbeda, dengan menetapkan teknik penelitian maka penelitian ini akan terarah sesuai dengan apa yang telah direncanakan.

2.4 Alat dan Bahan

Agar penelitian ini dapat dilakukan perlu adanya perangkat lunak dan perangkat keras jaringan komputer, dengan adanya perangkat tersebut penelitian dapat dilakukan dengan benar. Berikut adalah perangkat keras dan perangkat lunak yang digunakan dalam melakukan penelitian :

- 1) Kebutuhan Perangkat Keras
 - a. Menggunakan 4 unit router cisco 1841
 - b. Menggunakan 2 laptop, 1 sebagai server dan 1 client
 - c. Menggunakan *Cable UTP Cover Straight* dan *Cover Cross*
 - d. Menggunakan *Cable Console*
- 2) Kebutuhan Perangkat Lunak
 - a. Menggunakan aplikasi putty sebagai remote server,
 - b. Menggunakan aplikasi Axenettols, untuk menganalisis jaringan,
 - c. Membuat Sever Server FTP, sebagai media paket upload dan download.

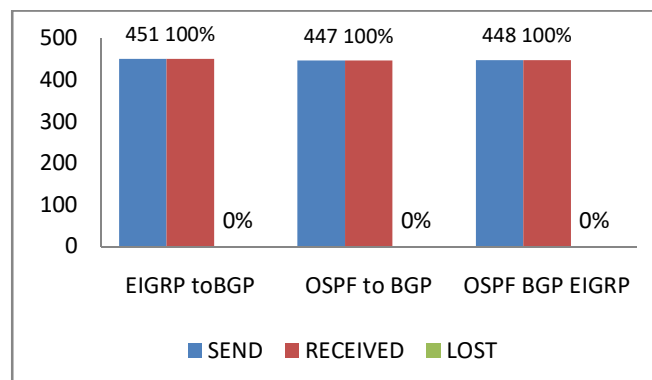
3. HASIL DAN PEMBAHASAN

Dalam melakukan perbandingan routing interior saat didistribusikan di-routing exterior penerimaan paket, bandwidth dan waktu pada masing-masing routing mendapatkan hasil yang berbeda. Berikut adalah hasil perbandingan pada masing-masing routing :

- 1) Dilihat dari paket yang diterima routing EIGRP menerima paket lebih besar dari routing OSPF.
- 2) Dilihat dari Pemakaian bandwidth routing EIGRP lebih banyak menggunakan bandwidth dari pada routing OSPF,
- 3) Dilihat dari pemakaian Waktu routing EIGRP memakan waktu lebih lama dari routing OSPF
- 4) Saat melakukan pengukuran keseluruhan dari Routing EIGRP to BGP, BGP to OSPF penerimaan paket, bandwidth dan pemakaian waktu semua routing baik internal maupun routing external masih dalam performa yang sangat baik.
- 5) Dilihat dari stabilitas jaringan, semua routing masih dalam katagori stabil hal ini disimpulkan masih termasuk kedalam katagori stabil dikarnakan tidak terdapat 1 paket data yang loss/ paket gagal.

Jika dilihat dari grafik pada masing-masing routing dinamis saat dilakukan redistribusi pada exterior gateway protokol, dapat dilihat di bawah ini :

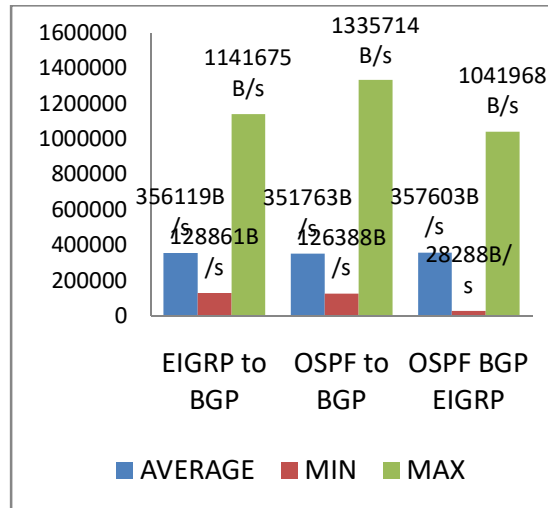
- 1) Grafik Perbandingan paket kirim dan terima pada masing masing routing yang sedang beraktifitas dapat di lihat di Gambar 1 di bawah ini :



Gambar 2 : Grafik Perbandingan Paket Pada masing-masing Routing

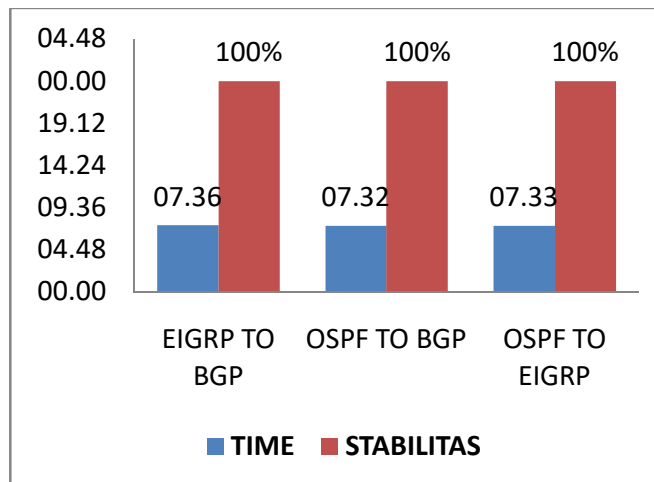
Dari gambar grafik diatas dapat dilihat routing EIGRP menerima paket lebih sedikit dibandingkan dengan routing OSPF, dan saat semua routing yang dianalisa di satukan ke-area Exterior paket tersebut bertambah banyak lagi. Hal ini dikarnakan saat routing EIGRP menuju routing OSPF.

2) Grafik Pemakaian Bandwidth pada masing masing routing saat jaringan beraktifitas dapat dilihat dibawah ini:



Gambar 3: Grafik Perbandingan Bandwidth Pada masing-masing Routing

3) Grafik Perbandingan durasi pemakaian waktu dan tingkat stabilitas jaringan pada saat jaringan beraktifitas dapat dilihat pada gambar 3 di bawah ini :



Gambar 4: Grafik Perbandingan Stability dan time

Dilihat dari hasil perbandingan pemakaian bandwidth, paket yang dikirim dan pemakaian waktu sudah jelas routing OSPF lebih baik dari pada routing EIGRP hal ini dikarnakan algortima masing-masing routing berbeda. Algoritma protokol yang digunakan oleh routing OSPF ialah menggunakan perhitungan djikstra, dimana algoritma ini adalah algortima yang mencari jalur terbaik dengan cara mengirim broadcast ke node tetangga sehingga mengetahui rute mana yang terbaik untuk mengatur paket yang dikirim agar sampai ke node tetangga . Sedangkan routing EIGRP, protokol routing ini menggunakan algortima Bellman-Ford dimana algortima ini menghitung atau mencari jalur terpendek untuk sampai ke node tujuan.

4. KESIMPULAN

Adapun kesimpulan yang dapat dipapar pada BAB V mengenai redistribusi routing interior to exterior pada jaringan komputer adalah sebagai berikut :

- 1) Dilihat dari pemakaian bandwidth routing OSPF memakai bandwidth lebih hemat dari pada routing EIGRP
- 2) Dilihat dari paket yang dipakai selama jaringan ber-operasi routing OSPF memakai paket lebih kecil dibandingkan dengan routing EIGRP
- 3) Dilihat dari waktu terima paket jaringan yang menggunakan routing
- 4) Dilihat dari tingkat stabilitas masing-masing routing masih dalam katagori stabil

Referensi

- Kristanto, B. (2012). *Aplikasi Algoritma Bellman-Ford dalam Meminimumkan Biaya Operasional Rute Penerbangan*. Doctoral dissertation. Yogyakarta: UIN SUNAN KALIJAGA.
- Purwananto, Y., Purwitasari, D. and Wibowo, A. (2005). *Implementasi dan Analisis Algoritma Pencarian Rute Terpendek di Kota Surabaya*. Jurnal Penelitian dan Pengembangan Telekomunikasi 10(2), pp.94-101.
- Yugianto, G.G. and Rachman, O. (2012). *Router: Teknologi, Konsep, Konfigurasi, dan Troubleshooting*. Bandung: Informatika.