

**Analisis Kinerja *Redistribution Routing*
Protokol OSPF, EIGRP dan BGP**

SKRIPSI

*Diajukan Untuk Memenuhi Salah Satu Syarat untuk Memperoleh
Gelar Sarjana Pada Fakultas Teknik
Universitas Islam Riau*



OLEH :

RISKI INDRA SAPUTRA

143510651

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS ISLAM RIAU
PEKANBARU
2021

LEMBAR PENGESAHAN PEMBIMBING SKRIPSI II

Nama : Riski Indra Saputra
NPM : 143510651
Jurusan : Teknik
Program Studi : Teknik Informatika
Jenjang Pendidikan : Strata Satu (S1)
Judul Skripsi : Analisis Kinerja *Redistribution Routing* Protokol OSPF, EIGRP dan BGP

Format sistematika dan pembahasan materi pada masing-masing bab dan sub bab dalam skripsi ini telah dipelajari dan dinilai relatif telah memenuhi ketentuan-ketentuan dan kriteria-kriteria dalam metode penelitian ilmiah. Oleh karena itu skripsi ini dinilai layak dapat disetujui untuk disidangkan dalam ujian **Seminar Komprehensif** .

Pekanbaru, 24 November 2021

Disetujui Oleh :

Ketua Prodi Teknik Informatika

Dosen Pembimbing

Dr. Apri Siswanto, S.Kom., M.Kom

Dr. Apri Siswanto, S.Kom., M.Kom

LEMBAR PENGESAHAN TIM PENGUJI UJIAN SKRIPSI

Nama : Riski Indra Saputra
NPM : 143510651
Jurusan : Teknik
Program Studi : Teknik Informatika
Jenjang Pendidikan : S1
Judul Skripsi : Analisis Kinerja *Redistribution Routing* Protokol OSPF, EIGRP dan BGP

Skripsi ini secara keseluruhan dinilai telah memenuhi ketentuan-ketentuan dan kaidah-kaidah dalam penulisan penelitian ilmiah serta telah diuji dan dapat dipertahankan dihadapan tim penguji. Oleh karena itu, Tim Penguji Ujian Skripsi Fakultas Teknik Universitas Islam Riau menyatakan bahwa mahasiswa yang bersangkutan dinyatakan **Telah Lulus Mengikuti Ujian Komprehensif pada tanggal 09 Desember 2021** dan disetujui serta diterima untuk memenuhi salah satu syarat guna memperoleh gelar Sarjana Strata Satu Bidang Ilmu **Teknik Informatika**.

Pekanbaru, 09 Desember 2021

Tim Penguji :

1. Dr. Ir. Evizal Abdul Kadir, M. Eng. Sebagai Tim Penguji I (.....)
2. Yudhi Arta, S.T., M.Kom Sebagai Tim Penguji II (.....)

Disahkan Oleh

Ketua Prodi Teknik Informatika

Dosen Pembimbing


Dr. Apri Siswanto, S.Kom., M.Kom


Dr. Apri Siswanto, S.Kom., M.Kom

LEMBAR PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan dibawah ini :

Nama : Riski Indra Saputra
NPM : 143510651
Fakultas : Teknik
Program Studi : Teknik Informatika
Jenjang Pendidikan : Strata-I (S1)

Dengan ini menyatakan dengan sesungguhnya bahwa skripsi yang saya tulis adalah benar dan asli hasil dari penelitian yang telah saya lakukan dengan judul **Analisis Kinerja Redistribution Routing Protokol OSPF, EIGRP dan BGP.**

Apabila dikemudian hari ada yang merasa dirugikan dan atau menuntut karena penelitian ini menggunakan sebagian hasil tulisan atau karya orang lain tanpa mencantumkan nama penulis yang bersangkutan, atau terbukti karya ilmiah ini bukan karya saya sendiri atau **plagiat** hasil karya orang lain, maka saya bersedia menerima sanksi sesuai dengan peraturan perundangan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sesungguhnya untuk dapat digunakan sebagaimana mestinya.

Pekanbaru, 09 Desember 2021

Yang membuat pernyataan,



Riski Indra Saputra

LEMBAR IDENTITAS PENULIS

Nama : Riski Indra Saputra
NPM : 143510651
Tempat/Tanggal Lahir : Batam, 16 Oktober 1994
Alamat Orang Tua : Sungai Piring, INHIL, Riau
Nama Orang Tua :
Nama Ayah : Muhammad Amin
Nama Ibu : Nenen Suharni
No.HP/Telp : 085271241287
Fakultas : Teknik
Program Studi : Teknik Informatika
Masuk Th.Ajaran : 2014
Keluar Th. Ajaran : 2021
Judul Penelitian : Analisis Kinerja Redistribution Routing Protokol
OSPF, EIGRP dan BGP

Pekanbaru, 09 Desember 2021

RISKI INDRA SAPUTRA

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Assalamu'alaikum warahmatullahi wabarakatuh...

Alhamdulillahirabbil'aalamiin, Allaahummashalli'ala Muhammad wa'ala aali Muhammad, penulis ucapkan atas nikmat, rahmat, karunia, kesehatan dan segala daya upaya yang Allah Ta'ala titipkan kepada penulis untuk menyelesaikan tulisan Tugas Akhir yang berjudul “ Analisa Kinerja *Redistribution Routing* Protokol OSPF, EIGRP dan BGP “ dalam rangka menyelesaikan studi Strata 1 Teknik Informatika Universitas Islam Riau.

Penulis juga mengucapkan terima kasih kepada semua pihak yang telah berkontribusi dalam penyelesaian tugas akhir ini, karena dengan dorongan motivasi dari mereka, penulis mampu menyelesaikan tugas akhir ini. Ucapan terimakasih ini penulis tujukan kepada :

1. Muhammad Amin (Ayah), Neneng Suharni (Ibu), Hj. Mastura (Nenek) dimana penulis tidak akan pernah mampu penulis balas jasa mereka sampai kapanpun.
2. H. Pajong Palompo (Paman), Hj. Masrita (Bibi) dimana penulis tidak akan pernah mampu penulis balas jasa mereka sampai kapanpun.
3. Bapak Dr. Eng Muslim selaku Dekan Fakultas Teknik Universitas Islam Riau

4. Bapak Dr. Apri Siswanto, ST, M. Kom, selaku pembimbing yang telah memberikan pengajaran, arahan dan telah sabar dalam memberikan bimbingan di sela – sela kesibukan beliau.
5. Bapak Dr. Apri Siswanto, ST, M. Kom, selaku Ketua Program Studi Teknik Informatika Universitas Islam Riau.
6. Bapak Dr. Ir. Evizal Abdul Kadir, M.Eng dan Bapak Yudhi Arta, ST, M. Kom, selaku dosen penguji skripsi penulis.
7. Seluruh Saudara – saudara penulis, yang dengan bantuan tenaga serta motivasi dari mereka terutama kepada Muhammad Fahlevi, Septa Dinda Nabella, Muhammad Fajrin dan Riska Debi Angelina selama perkuliahan.
8. Seluruh bapak dan ibu dosen program studi teknik informatika Universitas Islam Riau yang telah memberikan pembelajaran kepada penulis selama perkuliahan dari semester pertama hingga menyelesaikan studi Strata 1.
9. Staff prodi TI dan staff TU Teknik Universitas Islam Riau atas pelayanan administrasi penyelesaian Tugas Akhir ini.
10. Seluruh teman - teman seperjuangan angkatan 2014 khususnya kelas C.

Pekanbaru, 09 Desember 2021

Riski Indra Saputra

Analisis Kinerja *Redistribution Routing*

Protokol OSPF, EIGRP dan BGP

Riski Indra Saputra

Program Studi Teknik Informatika

Fakultas Teknik Universitas Islam Riau

Email : riskiinhil@student.uir.ac.id atau riskiindra234@gmail.com

ABSTRAK

Jaringan komputer sebuah teknologi yang dapat menghubungkan perangkat komunikasi yang tersedia pada zaman ini. Pada teknologi ini terdapat berbagai metode pendukungnya. Salah satunya *routing*. *Routing* merupakan sebuah metode agar sebuah jaringan dapat berkomunikasi dengan jaringan lainnya. Dalam *routing* dikenal sebuah istilah redistribusi ,yaitu melanjutkan kembali perjalanan paket ke jaringan dengan protokol berbeda. Pada penelitian ini, penulis menganalisa proses kinerja redistribusi antara OSPF dan BGP dengan EIGRP dan BGP berdasarkan nilai *throughput*, *packet loss* dan *delay*. Hasil dari analisa ini didapatkan kesimpulan bahwa redistribusi EIGRP dan BGP lebih efisien dibandingkan redistribusi OSPF dan BGP dari segi waktu. Hal ini terlihat dari waktu *delay* yang dimiliki redistribusi OSPF dan BGP dengan nilai 55,6 ms lebih besar dibandingkan redistribusi EIGRP dan BGP yaitu 3,6 ms.Selain itu pada redistribusi OSPF dan BGP terdapat paket komunikasi dengan nama *Hello Packets* dan *length* 90 dan pada redistribusi EIGRP dan BGP dengan nama *Hello* dan *length* 74.

Kata Kunci : Redistribusi, *Routing*, OSPF, BGP, EIGRP, *Throughput*, *Packet Loss*, *Delay*.

Redistribution Routing Performance Analysis

OSPF, EIGRP and BGP protocols

Riski Indra Saputra

Informatics Engineering Study Program

Faculty of Engineering, Islamic University of Riau

Email: riskiindhil@student.uir.ac.id or riskiindra234@gmail.com

ABSTRACT

Computer networking is a technology that can connect communication devices available at this time. Routing is one of the supporting methods on computer networks. Routing is a method so that a network can connect with other networks. In routing is known as redistribution, which is to re-travel packets to a network with a different protocol. In this study, the authors analyzed the redistribution performance process between OSPF and BGP with EIGRP and BGP based on throughput, packet loss and delay values. The results of this analysis concluded that EIGRP and BGP redistribution is more efficient than OSPF and BGP redistribution in terms of time. This is seen from the delay time that OSPF and BGP redistribution has with a value of 55.6 ms greater than the redistribution of EIGRP and BGP which is 3.6 ms. In addition to OSPF and BGP redistribution there is a communication package with the name Hello Packets and length 90 and on the redistribution of EIGRP and BGP with the name Hello and length 74.

Keywords : Redistribution, Routing, OSPF, BGP, EIGRP, Throughput, Packet Loss, Delay.

DAFTAR ISI

| | |
|--|-----|
| KATA PENGANTAR | i |
| ABSTRAK | iii |
| DAFTAR ISI | iv |
| DAFTAR TABEL | ix |
| DAFTAR GAMBAR | x |
| BAB I PENDAHULUAN | |
| 1.1 Latar Belakang | 1 |
| 1.2 Identifikasi Masalah..... | 2 |
| 1.3 Rumusan Masalah | 3 |
| 1.4 Tujuan Penelitian..... | 4 |
| 1.5 Batasan Masalah..... | 4 |
| 1.6 Manfaat Penelitian | 4 |
| BAB II LANDASAN TEORI | |
| 2.1 Tinjauan Pustaka | 6 |
| 2.2 Dasar Teori..... | 7 |
| 2.2.1 Pengertian <i>Routing</i> | 7 |
| 2.2.2 Jenis <i>Routing</i> | 8 |
| 2.2.3 Pengertian <i>Router</i> | 12 |
| 2.2.4 <i>Open Shortest Path First (OSPF)</i> | 14 |

| | | |
|----------------------------------|--|----|
| 2.2.5 | <i>Enhanced Interior Gateway Routing Protocol (EIGRP)</i> | 16 |
| 2.2.6 | <i>Border Gateway Protocol (BGP)</i> | 19 |
| 2.2.7 | <i>Redistribution</i> | 23 |
| 2.2.8 | <i>Transmission Control Protocol / Internet Protocol (TCP/ IP)</i> | 24 |
| 2.2.9 | Topologi Jaringan | 27 |
| 2.2.10 | <i>Flowchart</i> | 30 |
| 2.3 | Alat dan Bahan Penelitian | 32 |
| 2.3.1 | Perangkat Keras | 32 |
| 2.3.2 | Perangkat Lunak | 32 |
| 2.3.3 | Bahan Penelitian | 33 |
| 2.3.4 | Aplikasi PuTTY | 33 |
| 2.3.5 | Wireshark | 34 |
| BAB III METODE PENELITIAN | | |
| 3.1 | Metode Penelitian | 35 |
| 3.2 | Desain Topologi | 40 |
| 3.2.1 | Topologi Fisik | 40 |
| 3.2.2 | Topologi <i>Logical</i> | 42 |
| 3.3 | Perancangan Jaringan Menggunakan Protokol | 45 |
| 3.3.1 | Jaringan Protokol OSPF | 45 |
| 3.3.2 | Jaringan Protokol EIGRP | 48 |
| 3.3.3 | Jaringan Protokol BGP | 52 |
| 3.4 | Proses Redistribusi (<i>Redistribution</i>) Protokol..... | 56 |

| | |
|--|----|
| 3.4.1 Redistribusi Jaringan antara OSPF dan BGP | 57 |
| 3.4.2 Redistribusi Jaringan antara EIGRP dan BGP | 57 |

BAB IV HASIL DAN PEMBAHASAN

| | |
|---|----|
| 4.1 Pengujian Sistem | 59 |
| 4.2 Proses <i>Capture</i> paket ICMP menggunakan Wireshark..... | 59 |
| 4.2.1 Proses Mendapatkan Nilai <i>Throughput</i> | 61 |
| 4.2.2 Proses Mendapatkan Nilai <i>Packet Loss</i> | 62 |
| 4.2.3 Proses Mendapatkan Nilai <i>Delay</i> | 63 |
| 4.3 Skema Analisa <i>Throughput</i> , <i>Packet Loss</i> dan <i>Delay</i> Redistribusi OSPF dan BGP | 65 |
| 4.3.1 Proses Analisa Nilai <i>Throughput</i> Redistribusi OSPF dan BGP..... | 66 |
| 4.3.2 Proses Analisa Nilai <i>Packet Loss</i> Redistribusi OSPF dan BGP... 569 | |
| 4.3.3 Proses Analisa Nilai <i>Delay</i> Redistribusi OSPF dan BGP | 71 |
| 4.4 Skema Analisa <i>Throughput</i> , <i>Packet Loss</i> dan <i>Delay</i> Redistribusi EIGRP dan BGP | 74 |
| 4.4.1 Proses Analisa Nilai <i>Throughput</i> Redistribusi EIGRP dan BGP... 75 | |
| 4.4.2 Proses Analisa Nilai <i>Packet Loss</i> Redistribusi EIGRP dan BGP.... 78 | |
| 4.4.3 Proses Analisa Nilai <i>Delay</i> Redistribusi BGP dan EIGRP | 80 |
| 4.5 Perbandingan Nilai <i>Throughput</i> , <i>Packet Loss</i> , dan <i>Delay</i> Redistribusi BGP dan OSPF dengan Redistribusi BGP dan EIGRP..... | 82 |
| 4.5.1 Perbandingan Nilai <i>Throughput</i> antara Redistribusi OSPF dan BGP dengan Redistribusi EIGRP dan BGP. | 83 |
| 4.5.2 Perbandingan Nilai <i>Packet Loss</i> antara Redistribusi BGP dan OSPF dengan Redistribusi BGP dan EIGRP | 84 |

| | |
|--|----|
| 4.5.3 Perbandingan Nilai <i>Delay</i> antara Redistribusi OSPF dan BGP dengan Redistribusi EIGRP dan BGP. | 85 |
| 4.6 Kesimpulan Analisa Kinerja Redistribusi BGP dan OSPF dengan Redistribusi BGP dan EIGRP Berdasarkan Nilai <i>Throughput</i> , <i>Packet Loss</i> dan <i>Delay</i> | 85 |

BAB V KESIMPULAN DAN SARAN

| | |
|---------------------|----|
| 5.1 Kesimpulan..... | 90 |
| 5.2 Saran | 91 |

| | |
|-----------------------------|----|
| DAFTAR PUSTAKA | 93 |
|-----------------------------|----|



DAFTAR TABEL

| | |
|---|----|
| Tabel 2. 1 Simbol dan Keterangan Fungsi <i>Flowchart</i> | 30 |
| Tabel 4. 1 Nilai <i>Throughput</i> Redistribusi BGP dan OSPF..... | 69 |
| Tabel 4. 2 Nilai <i>Packet Loss</i> Redistribusi OSPF dan BGP..... | 71 |
| Tabel 4. 3 Nilai <i>Delay</i> Redistribusi OSPF dan BGP..... | 73 |
| Tabel 4. 4 Nilai <i>Throughput</i> Redistribusi BGP dan EIGRP..... | 77 |
| Tabel 4. 5 Nilai <i>Packet Loss</i> Redistribusi EIGRP dan BGP..... | 79 |
| Tabel 4. 6 Nilai <i>Delay</i> Redistribusi BGP dan EIGRP..... | 82 |
| Tabel 4. 7 Perbandingan Nilai <i>Throughput</i> antara Redistribusi BGP dan OSPF dengan Redistribusi BGP dan EIGRP..... | 83 |
| Tabel 4. 8 Perbandingan Nilai <i>Packet Loss</i> antara Redistribusi OSPF dan BGP dengan Redistribusi EIGRP dan BGP..... | 84 |
| Tabel 4. 9 Perbandingan Nilai <i>Delay</i> antara Redistribusi OSPF dan BGP dengan Redistribusi EIGRP dan BGP..... | 85 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 2. 1 Topologi <i>Bus</i> | 28 |
| Gambar 2. 2 Topologi <i>Tree</i> | 28 |
| Gambar 2. 3 Topologi <i>Ring</i> | 29 |
| Gambar 2. 4 Topologi <i>Mesh</i> | 29 |
| Gambar 2. 5 Topologi <i>Star</i> | 30 |
| Gambar 3. 1 Pencarian Nilai <i>Throughput</i> pada Wireshark..... | 38 |
| Gambar 3. 2 Pencarian Nilai <i>Packet Loss</i> pada Wireshark. | 39 |
| Gambar 3. 3 Pencarian Nilai <i>Delay</i> pada Wireshark..... | 39 |
| Gambar 3. 4 Topologi Fisik Redistribusi OSPF dan BGP. | 41 |
| Gambar 3. 5 Topologi Fisik Redistribusi EIGRP dan BGP. | 42 |
| Gambar 3. 6 Topologi <i>Logic</i> Redistribusi OSPF dan BGP. | 43 |
| Gambar 3. 7 Topologi <i>Logic</i> Redistribusi EIGRP dan BGP. | 44 |
| Gambar 3. 8 Topologi Jaringan Protokol OSPF..... | 45 |
| Gambar 3. 9 Perangkat Nyata Laptop Klien pada OSPF. | 46 |
| Gambar 3. 10 Konfigurasi IP Interface pada Router_OSPF. | 46 |
| Gambar 3. 11 Slot GE0/0 dan GE0/1 pada Router_OSPF. | 47 |
| Gambar 3. 12 Konfigurasi Protokol OSPF pada Router_OSPF..... | 47 |
| Gambar 3. 13 Ip <i>Interface</i> pada Router_Redistribusi..... | 48 |
| Gambar 3. 14 Konfigurasi Protokol OSPF pada Router_Redistribusi. | 48 |
| Gambar 3. 15 Jaringan Topologi Protokol EIGRP..... | 49 |

| | |
|---|----|
| Gambar 3. 16 Perangkat Nyata Laptop Klien pada EIGRP. | 49 |
| Gambar 3. 17 Konfigurasi IP Interface Router_EIGRP. | 50 |
| Gambar 3. 18 Slot GE0/0 dan GE0/1 pada Router_EIGRP. | 50 |
| Gambar 3. 19 Konfigurasi Protokol EIGRP pada Router_EIGRP. | 51 |
| Gambar 3. 20 Ip Interface Pada GE0/1 Router_Redistribusi. | 51 |
| Gambar 3. 21 Konfigurasi Protokol EIGRP pada Router_Redistribusi. | 52 |
| Gambar 3. 22 Topologi Jaringan Protokol BGP. | 53 |
| Gambar 3. 23 Laptop Acer sebagai <i>Server</i> di Jaringan BGP. | 53 |
| Gambar 3. 24 Konfigurasi IP Interface pada Router_BGP. | 54 |
| Gambar 3. 25 Slot GE0/0 dan GE0/1 pada Router_BGP. | 54 |
| Gambar 3. 26 Konfigurasi Protokol BGP pada Router_BGP. | 55 |
| Gambar 3. 27 Ip <i>Interface</i> pada Router_Redistribusi. | 55 |
| Gambar 3. 28 Konfigurasi Protokol BGP pada Router_Redistribusi. | 56 |
| Gambar 3. 29 Redistribusi antara OSPF dan BGP. | 57 |
| Gambar 3. 30 Redistribusi antara EIGRP dan BGP. | 58 |
| Gambar 4. 1 Proses <i>PING</i> | 60 |
| Gambar 4. 2 Proses <i>Capture</i> Paket ICMP pada Wireshark. | 60 |
| Gambar 4. 3 Proses Mendapatkan Nilai <i>Throughput</i> | 61 |
| Gambar 4. 4 <i>Ping Packet Loss</i> 0%. | 62 |
| Gambar 4. 5 <i>Ping Packet Loss</i> 100%. | 63 |
| Gambar 4. 6 Proses Mendapatkan Nilai <i>Delay</i> | 64 |
| Gambar 4. 7 <i>Form File Properties</i> | 65 |

| | |
|---|----|
| Gambar 4. 8 Topologi Jaringan Redistribusi OSPF dan BGP..... | 66 |
| Gambar 4. 9 Koneksi Perangkat Sumber menuju Perangkat Tujuan. | 67 |
| Gambar 4. 10 <i>Ping</i> dari Laptop di OSPF (Windows 10) menuju <i>Server</i> di BGP. | 67 |
| Gambar 4. 11 Penangkapan Paket ICMP oleh Wireshark pada Redistribusi OSPF dan BGP..... | 68 |
| Gambar 4. 12 Tampilan <i>Capture File Properties</i> 1 kali <i>Ping</i> pada | 68 |
| Gambar 4. 13 Tampilan <i>Packet Loss</i> yang dihasilkan <i>Ping</i> Pada | 70 |
| Gambar 4. 14 Nilai <i>Packet Loss</i> dari Hasil <i>Capture File Properties</i> | 70 |
| Gambar 4. 15 Tampilan Hasil <i>Capture Ping</i> ke-1..... | 71 |
| Gambar 4. 16 Tampilan File csv Hasil Capture ICMP Ping ke-1 Redistribusi OSPF dan BGP..... | 72 |
| Gambar 4. 17 Topologi Jaringan Redistribusi EIGRP dan BGP..... | 74 |
| Gambar 4. 18 Koneksi Perangkat Sumber menuju Perangkat Tujuan. | 75 |
| Gambar 4. 19 <i>Ping</i> dari Laptop di EIGRP (Windows 10) menuju <i>Server</i> di BGP. .. | 76 |
| Gambar 4. 20 Penangkapan Paket ICMP oleh Wireshark pada Redistribusi EIGRP dan BGP..... | 76 |
| Gambar 4. 21 Tampilan <i>Capture File Properties</i> 1 kali <i>Ping</i> Pada | 77 |
| Gambar 4. 22 Tampilan <i>Packet Loss</i> yang dihasilkan <i>Ping</i> Pada | 78 |
| Gambar 4. 23 Nilai <i>Packet Loss</i> dari Hasil <i>Capture File Properties</i> | 79 |
| Gambar 4. 24 Tampilan Hasil Capture <i>Ping</i> Redistribusi..... | 80 |
| Gambar 4. 25 Tampilan File csv Hasil Capture ICMP <i>Ping</i> Redistribusi EIGRP dan BGP. | 81 |

| | |
|---|----|
| Gambar 4. 26 Perbandingan Nilai <i>Packet Loss</i> | 86 |
| Gambar 4. 27 Perbandingan Nilai <i>Throughput</i> | 86 |
| Gambar 4. 28 Perbandingan Nilai <i>Delay</i> | 87 |
| Gambar 4. 29 Tampilan Wireshark Redistribusi OSPF dan BGP | 88 |
| Gambar 4. 30 Tampilan Wireshark Redistribusi EIGRP dan BGP | 88 |
| Gambar 4. 31 Perbedaan paket OSPF dan EIGRP | 89 |



BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi komputer saat ini semakin canggih. Dengan teknologi komputer, pengelolaan informasi menjadi lebih baik dan efisien, baik itu dari segi waktu, jarak dan biaya. Dalam teknologi komputer, terdapat beberapa hal yang menunjang teknologi tersebut, diantaranya teknologi hardware, software, dan jaringan komputer. Ketiga hal tersebut menjadi penunjang kinerja teknologi komputer dalam pengelolaan informasi.

Jaringan komputer merupakan sebuah teknologi yang membuat *software* maupun *hardware* dapat saling berkomunikasi satu sama lain. Dalam jaringan komputer terdapat teknologi yang disebut *routing*. *Routing* merupakan sebuah proses untuk meneruskan paket-paket jaringan dari satu jaringan ke jaringan lainnya melalui sebuah antar-jaringan.

Routing juga dapat merujuk kepada sebuah metode penggabungan beberapa jaringan sehingga paket-paket data dapat diantarkan dari satu jaringan ke jaringan selanjutnya. Penerapan teknologi *routing* pada sebuah jaringan komputer terdapat berbagai macam, namun yang banyak digunakan adalah *routing* statis dan *routing* dinamis.

Pada *routing* statis, proses pengaturan *router* jaringan menggunakan tabel *routing* secara manual. Sedangkan *routing* dinamis, *router* memiliki kemampuan untuk membuat tabel *routing* secara otomatis berdasarkan lalu lintas jaringan dari

router yang terhubung. *Router* merupakan sebutan untuk alat yang digunakan dalam proses *routing*.

Pada penelitian ini, pokok pembahasan adalah *routing* dinamis. Pada *routing* dinamis terjadi perkembangan teknologi protokol yang digunakan, sehingga *routing* dinamis terbagi beberapa macam jenis protokol, diantaranya, *Routing Information Protocol* (*RIP*), *Interior Gateway Routing Protocol* (*IGRP*), *Open Short Path First* (*OSPF*), *Enhanced Interior Gateway Routing Protocol* (*EIGRP*), *Border Gateway Protocol* (*BGP*). Dalam penerapan *routing* protokol dinamis pada sebuah topologi jaringan, terdapat sebuah metode yang disebut dengan *redistribution*.

Redistribution atau dalam serapan Bahasa Indonesia disebut dengan redistribusi merupakan sebuah metode *routing* protokol yang digunakan untuk mendistribusikan atau meneruskan suatu *routing* protokol ke routing protokol lain (protokol yang berbeda) agar dapat saling menukarkan *routing table* masing-masing dengan tujuan agar *routing* protokol yang berbeda tersebut bisa berkomunikasi.

Pada penelitian ini, penulis membatasi penelitian terhadap kualitas *routing* berdasarkan nilai *delay*, *throughput* dan *packet loss* dalam redistribusi *routing* antara redistribusi protokol BGP dengan OSPF dan BGP dengan EIGRP.

1.2 Identifikasi Masalah

Berdasarkan uraian latar belakang yang dijelaskan sebelumnya, penulis mengidentifikasi beberapa masalah pada penelitian ini, diantaranya, :

1. OSPF, EIGRP dan BGP merupakan protokol yang berbeda sehingga memiliki

aturan, kriteria dan konfigurasi yang berbeda tetapi pada penelitian ini, topologi jaringan dari 3 protokol tersebut harus bisa berkomunikasi.

2. Perlu sebuah metode agar topologi jaringan dengan protokol OSPF, EIGRP dan BGP tersebut bisa berkomunikasi, dan pada penelitian ini penulis menerapkan metode *routing* redistribusi.
3. Melakukan analisa terhadap kinerja redistribusi antara protokol OSPF, EIGRP dan BGP dengan skema redistribusi protokol BGP diredistribusi ke OSPF dan BGP diredistribusi ke EIGRP berdasarkan nilai *throughput*, *packet loss* dan *delay*.

1.3 Rumusan Masalah

Berdasarkan uraian identifikasi masalah yang telah disebutkan, maka pada penelitian ini dapat diambil sebuah rumusan masalah. Adapun rumusan masalah tersebut diantaranya :

1. Bagaimana kinerja *routing* dinamis pada sebuah topologi jaringan berdasarkan redistribusi protokol BGP dengan OSPF dan redistribusi BGP dengan EIGRP, berdasarkan nilai *throughput*, *packet loss* dan *delay* ?
2. Redistribusi protokol manakah yang lebih baik dalam kinerja *routing* dinamis antara BGP redistribusi dengan OSPF dan BGP redistribusi dengan EIGRP berdasarkan nilai *throughput*, *packet loss* dan *delay* ?

1.4 Tujuan Penelitian

Dalam melakukan penelitian ini, penulis ingin mengetahui hasil dari penelitian yang dilakukan dan mampu mencapai beberapa tujuan sesuai dengan masalah yang diteliti. Adapun tujuan penelitian ini diantaranya, :

1. Mengetahui perbedaan kinerja *routing* redistribusi protokol OSPF dan BGP dengan BGP dan EIGRP berdasarkan nilai *throughput*, *packet loss* dan *delay* .
2. Mengetahui mana kinerja yang lebih efisien antara redistribusi protokol BGP dan OSPF dengan redistribusi protokol BGP dan EIGRP berdasarkan nilai *throughput*, *packet loss* dan *delay*.

1.5 Batasan Masalah

Pada analisis kinerja *redistribution routing* protokol OSPF, EIGRP dan BGP, penulis membuat batasan masalah diantaranya, :

1. Penulis hanya menggunakan protokol OSPF, EIGRP dan BGP sebagai protokol perbandingan.
2. Skema peredistribusian untuk melakukan perbandingan adalah redistribusi protokol BGP dengan OSPF dan redistribusi BGP dengan EIGRP.
3. Menggunakan paket ICMP sebagai alat komunikasi antar perangkat.
4. Proses analisa yang dilakukan dengan melakukan penangkapan paket ICMP dan menghitung nilai *delay*, *throughput* dan *packet loss* dari paket ICMP tersebut.

1.6 Manfaat Penelitian

Adapun manfaat dari penelitian ini diantaranya, :

1. Mengetahui perbedaan kinerja *routing* redistribusi protokol BGP dengan

OSPF dan redistribusi BGP dengan EIGRP, dengan membangun sebuah jaringan komputer berdasarkan simulasi dengan perangkat nyata berdasarkan nilai *throughput*, *packet loss* dan *delay*.

2. Dapat menjadi pertimbangan penerapan metode *redistribution* dalam *routing* dinamis, *redistribution* protokol manakah antara BGP dengan OSPF, dan BGP dengan EIGRP berdasarkan nilai *throughput*, *packet loss* dan *delay*.



Dokumen ini adalah Arsip Miik :

Perpustakaan Universitas Islam Riau

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian yang dijadikan rujukan dalam penelitian ini antara lain, yang pertama adalah penelitian yang dilakukan Dwi Aryanta dan Bayu Agung Pranata (2014) dalam jurnalnya dengan judul - Perancangan dan Analisis *Redistribution Routing Protocol* OSPF dan EIGRP, mereka melakukan pengujian kinerja *routing* dengan membandingkan topologi jaringan yang menerapkan protokol tunggal dan penggabungan protokol.

Pada protokol tunggal, mereka menganalisa kinerja *routing* protokol OSPF to OSPF dan EGRP to EIGRP. Sedangkan untuk penggabungan protokol, mereka menerapkan metode *redistribution* dalam satu topologi jaringan dengan menggunakan protokol OSPF dan EIGRP.

Penelitian selanjutnya adalah penelitian yang dilakukan Yoldi Novendra, Yudhi Arta dan Apri Siswanto (2018) dalam jurnal mereka yang berjudul — Analisis Kinerja *Routing* OSPF dan EIGRP. Pada penelitian ini mereka melakukan perbandingan kinerja *routing* protokol OSPF dan EIGRP berdasarkan parameter QoS, yaitu *Bandwith*, *Delay* dan *Packet Loss*.

Berikutnya adalah penelitian yang dilakukan Akhmad Zulfikar Al Ghivani (2018) dalam jurnalnya yang berjudul Perbandingan *Routing* Protokol BGP dan EIGRP, Evaluasi Kinerja Performansi pada *Autonomous System* (AS) berbeda. Beliau menarik kesimpulan bahwa efisiensi kinerja BGP sangat baik karena

menggunakan algoritma *link-state*, dimana BGP mengenali jaringan tetangganya menggunakan atribut seperti *bandwidth* pada jaringannya. Sementara EIGRP sedikit lebih baik karena adanya kombinasi *link - state* dan *distance vector*, di mana selain berdasarkan atribut, EIGRP mengenai tetangganya yang terdekat berdasarkan jalur terpendek yang langsung terhubung dengan *router* tujuan.

Sebagai bahan tambahan rujukan adalah penelitian yang dilakukan Atif Manzoor, Muzammil Hussain dan Sobia Mehrban (2019) dalam jurnal mereka yang berjudul — *Performance Analysis and Route Optimization: Redistribution between EIGRP, OSPF & BGP Routing Protocols*, berdasarkan penelitian ini mereka mengambil kesimpulan bahwa EIGRP jauh lebih cepat dalam konvergensi dan mengambil paling sedikit waktu untuk menyatukan perubahan topologi dari pada OSPF dan BGP.

Pada penelitian yang dilakukan Dimas (2014), dengan membandingkan protokol EIGRP dan OSPF yang diimplementasikan dalam topologi *ring* dan *mesh* didapat kesimpulan bahwa protokol EIGRP lebih baik dibandingkan OSPF untuk setiap topologi. Pada penelitian ini, menekankan perbandingan *time delay* dan perhitungan *trace route*. Untuk pemilihan *trace route* berdasarkan nilai *metric* dan *cost* terkecil semua *routing*.

2.2 Dasar Teori

2.2.1 Pengertian *Routing*

Routing merupakan teknologi yang memproses penentuan jalur dan pengiriman paket data antar *network* yang berbeda. *Routing* melakukan proses pengiriman data antar *network* yang berbeda dengan cara mem-*forward* paket

data melalui *gateway*. *Routing* menentukan kemana datagram akan dikirim agar mencapai tujuan yang diinginkan (M.Rizqiariadi, 2014).

Menurut Hasanah dan Mubarakah (2014), *Routing* merupakan sebuah proses yang terjadi dalam jaringan komputer untuk meneruskan paket – paket antar jaringan komputer yang berbeda sehingga membentuk satu rute tertentu. Dalam proses ini diperlukan alat yang dinamakan *router*. Ada 5 poin informasi yang dibutuhkan *router* dalam melakukan *routing* yaitu:

1. Alamat tujuan / *destination address*.
2. Mengenal sumber informasi.
3. Menemukan rute.
4. Pemilihan rute.
5. Menjaga informasi *routing*.

Proses *routing* diperlukan karena *router-router* dalam jaringan hanya mengatur proses pengiriman paket data ke jalur yang terhubung tidak langsung, setiap *router* dalam jaringan tidak mempelajari jalur yang terhubung langsung dengannya. *Routing* harus memiliki kemampuan mengatasi perubahan topologi jaringan dan juga lalu lintas jalur tanpa membatalkan proses pada *host*. Selain itu, jika *router* mengalami tabrakan pada proses pengiriman paket data, jaringan tidak melakukan *reboot*.

2.2.2 Jenis *Routing*

Routing terbagi dalam 2 jenis yaitu *routing* statis dan *routing* dinamis. Kedua jenis *routing* tersebut memiliki kelebihan dan kekurangan masing – masing. Berikut ini merupakan penjelasan *routing* statis dan *routing* dinamis.

A. *Routing* Statis

Routing statis merupakan sebuah proses peroutingan yang bersifat statis. Proses setiap konfigurasi dilakukan secara manual oleh *administrator* jaringan. Pada *routing* statis, *router* tidak berbagi informasi *routing*, sehingga apabila terjadi penambahan (perubahan) sebuah *network* kedalam topologi jaringan *routing* statis, *administrator* harus menambahkan sebuah *routing table* baru ke semua *router* yang terhubung dalam topologi jaringan tersebut dengan tujuan *network* baru tersebut dapat dikenali oleh *network* yang ada.

Menurut Erma Julita, dkk (2017), *routing* statis adalah pengaturan *routing* jaringan secara manual yang dilakukan oleh *administrator* jaringan. *Admin* membuat tabel *routing* yang akan dilalui oleh paket data dalam jaringan berupa *ip address* yang akan dilalui data. Rute pada *routing* statis tidak akan berubah, kecuali diubah secara manual oleh *admin*. Untuk skala kecil *routing* statis bukanlah hal yang sulit untuk diimplementasikan.

Cara kerja *routing* statis dibagi menjadi 3 bagian diantaranya,

1. Jaringan yang mengkonfigurasi *router*, artinya jika terjadi perubahan *network* pada sebuah topologi jaringan *routing* statis, maka setiap *router* akan dikonfigurasi ulang oleh *administrator* jaringan.
2. *Router* melakukan *routing* berdasarkan informasi dalam tabel *routing*.
3. *Routing* statis digunakan untuk melewati paket.

Keuntungan *routing* statis, diantaranya :

- Beban kinerja *router* pada *routing* statis lebih ringan dibanding pada *routing* dinamis. Karena *router* pada *routing* statis hanya melakukan

update ip tabel *routing* hanya sekali saja, yaitu pada saat *administrator* melakukan konfigurasi.

- Pengiriman paket data lebih cepat karena jalur sudah ditentukan dan diketahui terlebih dahulu.
- Dapat dengan cepat mengetahui kesalahan atau penyebab *troubleshoot* pada jaringan.
- Jalur *routing* mudah diprediksi.
- Proses *update routing table* tidak dibutuhkan.
- Mudah dikonfigurasi untuk *network* kecil.

Kerugian pada *routing* statis, diantaranya :

4. Tidak bisa digunakan untuk skala jaringan yang besar.
5. Jika terjadi penambahan *router*, maka semua konfigurasi tiap *router* terdahulu harus dirubah.
6. Tidak bisa melakukan penyesuaian jika muncul *link failure* pada salah satu jalur.
7. Alamat ip jaringan yang akan dikenalkan atau dituju harus diketahui beserta *next hoopnya*

B. *Routing* Dinamis

Routing dinamis merupakan proses *routing* dengan menggunakan *routing* protokol dimana tabel *routing* bekerja secara otomatis. *Routing* dinamis dibagi 2 kategori yaitu, *distance vector* dan *link state routing* protokol (Erma Julita, dkk 2017). Menurut Yovie Dwi Villasica dan Naemah Mubarakah (2014), *routing* dinamis merupakan *routing* yang merutekan jalur secara otomatis, dimana jika

terjadi perubahan topologi antar jaringan, maka *router* secara otomatis akan membuat *routing* yang baru.

Fungsi dasar dari *routing* dinamis adalah memberikan fitur pada *router* agar secara otomatis mampu membuat tabel *routing* pada saat terjadi perubahan dalam suatu jaringan komputer. *Routing* dinamis memiliki *routing* protokol yang memiliki fungsi untuk memberikan jalur terbaik dan memperbarui informasi tabel *routing* ketika terjadi perubahan jaringan.

Menurut Wood, dkk (2015) dan Circiumarescu, dkk (2015), pada *routing* dinamis terdapat 2 klasifikasi protokol yakni :

1. *Interior Gateway Protocol* (IGP) digunakan untuk *routing* dalam *autonomous system*. Contohnya EIGRP.
2. *Exterior Gateway Protocol* (EGP) digunakan untuk *routing* antara *autonomous system* yang berbeda. Contohnya BGP.

Ada beberapa jenis *routing* dinamis yang banyak digunakan diantaranya *Routing Information Protocol* (RIP), *Open Shortest Path First* (OSPF), *Border Gateway Protocol* (BGP), *Interior Gateway Routing Protocol* (IGRP) dan *Enhanced Interior Gateway Routing Protocol* (EIGRP). Berbagai macam jenis *routing* tersebut memiliki kelebihan dan kekurangannya masing-masing yang mana tidaklah sempurna (Masykur, 2016), oleh karena itu terkadang *traffic* perlu ditentukan dan diprediksi secara manual.

Kelebihan *routing* dinamis, diantaranya :

1. Hanya mengenalkan alamat yang terhubung langsung dengan *routernya*.
2. Tidak perlu mengetahui semua alamat *network* yang ada.

3. Bila terjadi penambahan suatu *network* baru, tidak perlu semua *router* dikonfigurasi. Hanya *router – router* yang berkaitan.
4. *Adaptability*, yaitu rute dapat berubah secara otomatis dalam penyesuaian kondisi tertentu, misal adanya *link failure*.

Kekurangan *routing* dinamis diantaranya, :

1. Beban kerja *router* lebih berat karena selalu memperbarui *ip table* pada setiap waktu tertentu.
2. Kecepatan pengenalan dan kelengkapan *ip table* terbilang lama karena *router* melakukan *broadcast* ke semua *router* sampai ada yang cocok. Sehingga setelah konfigurasi harus menunggu beberapa saat agar setiap *router* mendapat semua alamat *ip* yang ada.
3. Kompleksitas algoritma *routing* meningkat. *Router* menentukan rute berdasarkan, misalnya: *bandwidth* yang tersedia, jalur terpendek, dll.
4. *Router* harus saling bertukar informasi *routing* secara periodik.
5. Tidak semua *router* mendukung *dynamic routing*.

2.2.3 Pengertian Router

Istilah *router* pada dasarnya berasal dari Bahasa Inggris yaitu kata *Route* yang artinya trayek atau mengarahkan perjalanan melalui yang ditambahi dengan imbuhan *-er* yang maknanya pelaku. Sehingga, jika diartikan dari asal katanya, maka *router* dapat diartikan sebagai sebuah perangkat yang digunakan untuk mengarahkan sinyal pada jaringan komputer.

Menurut O'brien (2006: 193), lebih spesifik menyatakan bahwa, *router* adalah sebuah alat jaringan. Komputer yang mengirimkan paket data melalui

sebuah jaringan atau internet menuju tujuannya, melalui sebuah proses yang dikenal sebagai *routing*. Sedangkan menurut Hikmaturokhman, dkk (2010), supaya suatu paket dapat mencapai tujuannya, diperlukan suatu peralatan untuk mengatur paket - paket tersebut agar mencapai tujuannya dengan jalan yang tersingkat. Untuk itu perlu adanya *router* yang fungsi utamanya adalah untuk menentukan jalur dan mengirimkan paket-paket dari suatu jaringan ke jaringan lain. Agar *router* dapat mengetahui jalur mana yang terbaik untuk mengirimkan paket ke alamat yang dituju, *router* menggunakan peta atau tabel *routing*.

Pada dasarnya fungsi *router* hampir sama dengan fungsi *switch* yaitu sebagai perangkat penghubung dalam jaringan komputer. Perangkat *router* digunakan untuk menghubungkan satu jaringan komputer dengan jaringan komputer lainnya.

Jenis – jenis *router* diantaranya sebagai berikut :

A. *Router Aplikasi*

Pengertian *router* aplikasi adalah sebuah aplikasi (*software*) yang digunakan untuk menanamkan kemampuan perangkat *router* pada perangkat komputer biasa. Beberapa contoh *router* aplikasi yang banyak digunakan oleh para pengguna komputer yaitu WinGate, Winroute, SpyGate, WinProxy, dan lain – lain.

B. *Router Hardware*

Router hardware merupakan sebuah perangkat keras yang memiliki kemampuan dasar untuk menangkap dan menyebarkan sinyal. *Router hardware* biasanya lebih dikenal dengan istilah *router* saja. Berbeda dengan *router* aplikasi

yang tidak memiliki bentuk fisik sama sekali, *router hardware* memiliki bentuk fisik yang nyata layaknya perangkat keras lainnya.

C. Router PC

Router PC merupakan sebuah perangkat personal komputer biasa yang dimodifikasi sehingga dapat dipakai untuk menyebarkan sinyal layaknya sebuah *router*. *Router* PC dapat dibuat dari perangkat komputer berspesifikasi rendah sekalipun (minimal *spec.* Pentium dua, *hard drive* 10 GB dengan ram 64 mb). Untuk menjadikan perangkat komputer menjadi sebuah *router*, perangkat komputer perlu diinstal sistem operasi khusus *router* seperti salah satunya sistem operasi Mikrotik. Fungsi utama *router* adalah melewatkan data antar segmen yang memiliki alamat *network* yang berbeda (Musril, 2015).

2.2.4 Open Shortest Path First (OSPF)

Open Shortes Path First (OSPF) adalah protokol *routing* dinamis yang merupakan bagian dari *interior protocol routing*. OSPF didesain oleh *Internet Engineering Task Force* (IETF). OSPF merupakan pengembangan dari algoritma *Shortest Path First* (SPF) dan protokol ini juga merupakan *routing* protokol yang berstandar terbuka, yaitu *routing* protokol ini bukan ciptaan dari vendor manapun. OSPF dipublikasikan pada RFC nomor 1247. OSPF bekerja berdasarkan algoritma *Shortest Path First* yang dikembangkan berdasarkan algoritma Dijkstra sebagai *Interior Gateway protocol* (IGP).

IGP dikembangkan untuk menghubungkan *router - router* dibawah kendali *administrator* jaringan (Sofana, 2008). Menurut Gerry (2012), OSPF merupakan *routing* protokol yang menggunakan teknologi *link State* yang memang didesain

untuk bekerja dengan sangat efisien dalam proses pengiriman *update* informasi rute. Hal ini menjadi sangat cocok untuk dikembangkan menjadi *network* berskala besar. OSPF memiliki beberapa karakteristik diantaranya :

- Merupakan protokol *link-state* dimana dapat memelihara rute dalam dinamik network struktur dan dapat dibangun beberapa bagian dari *subnetwork*.
- Menggunakan algoritma SPF untuk menghitung *cost* terendah.
- *Date routing* dilakukan secara *flooded* saat terjadi perubahan topologi jaringan.

OSPF memiliki titik berat dalam hal kinerja *processor*, penggunaan *bandwith* dan kebutuhan ruang memori. Hal ini disebabkan OSPF merupakan protokol *link-state* dimana kelemahan dari *link-state* adalah membutuhkan *resource* yang besar.

Berikut adalah proses algoritma djikstra pada teknik OSPF, diantaranya :

1. Pertama data masih berupa bit-bit yang kemudian dikemas menjadi paket-paket data.
2. Setiap *router* mengupdate tabel *routing*.
3. Setelah mengupdate tabel *routing*, *router* memberitahukan kepada *router* tetangganya berapa jarak dari *router* tersebut.
4. Tiap *router* mendapatkan informasi dari tabel *routing* yang telah di *update* lalu algoritma djikstra menghitung semua jarak yang menuju alamat tujuan.
5. Algoritma mencari *best path* ke alamat tujuan,jika alamat yang dituju

sudah ketemu maka tiap *router* mengupdate informasi pada tabel *routing*, jika tidak maka algoritma akan menghitung ulang untuk mencari *best path*.

6. Jika semua *router* telah di periksa maka proses selesai, jika belum, maka *router* akan memberi tahu kembali *router* tetangganya.

2.2.5 Enhanced Interior Gateway Routing Protocol (EIGRP)

Enhanced Interior Gateway Routing Protocol (EIGRP) merupakan salah satu jenis protokol yang ada pada *routing* dinamis. EIGRP merupakan protokol *routing* yang dikembangkan oleh cisco.

Protokol ini merupakan hasil pengembangan dari protokol sebelumnya, yaitu IGRP. EIGRP masih dapat menerima *network* lama yang menggunakan *router – router* IGRP, dan begitu pula sebaliknya. Menurut cisco, EIGRP memiliki 4 kriteria yang membedakannya dari protokol lain, diantaranya :

1. Mendukung *fast convergence*.
2. Mendukung *variable length subnet mask* (VLSM).
3. Mendukung *partial updates*.
4. Mendukung *multiple network layer protocols*.

EIGRP adalah protokol *hybrid*, yang artinya protokol ini merupakan gabungan dari *distance vector routing* protokol dan *linked state routing* protokol. Algoritma yang digunakan protokol EIGRP adalah *diffusing update algorithm* (DUAL) dimana hal ini berdasarkan penjelasan Dumitrache (2017), dimana dia mengatakan bahwa EIGRP memiliki algoritma alternatif (*link-state* dan *distance vector*) sehingga untuk menentukan jalur terbaik dapat lebih cepat diproses

daripada OSPF yang hanya berdasarkan *link - state*. Hal membuat semua *router* tersinkronisasi di dalam jaringannya. EIGRP memiliki konsep *successor* dan *feasible successor* dalam *routing* tabelnya, *successor* adalah *link* utama sedangkan *feasible successor* adalah cadangan ketika *Link* utama putus.

Menurut Al-Musawi (2017), EIGRP memiliki sistem pembuatan *routing* protokol dengan cara memastikan dan menyediakan jaringan yang sudah dipastikan lingkungan *loopless* routingsnya, dengan demikian maka dalam *router* tidak akan terjadi perputaran data ketika salah satu *agent* mati.

Protokol *routing* EIGRP memiliki karakteristik sebagai berikut :

1. Merupakan sebuah *balanced hybrid routing protocol*. Melakukan *advertising* rute secara langsung kepada *neighbors*. Kondisi ini mirip protokol *distance vector*. Menggunakan sekumpulan *table/database* yang mirip dengan database *link state*.
2. *Advertise* menggunakan alamat *multicast* (224.0.0.10).
3. Menggunakan *Diffusing Update Algorithm* (DUAL) untuk menentukan *path* terbaik.
4. Waktu *Convergence* yang sangat cepat ketika menjumpai ada *link* yang gagal.
5. Mendukung *equal cost load balance* dan *unequal cost load balance*.
6. Mendukung VLSM.
7. Parameter yang digunakan untuk menentukan rute terbaik :
 - a. *Advertised Distance* (AD).
 - b. *Feasible Distance* (FD).

8. Menggunakan perhitungan *sophisticated metric*.
9. Mendukung *authentication*.
10. Mendukung manual *summarization* pada semua *interface*.
11. Bekerja pada IP protocol 88. EIGRP menyediakan 5 buah *message package*, diantaranya :

- a. *Hello*.
- b. *Update*, (bersifat *reliable*).
- c. *Query*, (bersifat *reliable*).
- d. *Reply*, (bersifat *reliable*).
- e. *ACK*.

12. EIGRP bersifat *reliable*, namun *ACK* dan *hello* tidak di-*acknowledg*.

13. EIGRP menyediakan 3 buah tabel:

- a. *Neighbor table*.
- b. *Topology table*.
- c. *IP Routing table*.

14. Fungsi EIGRP dikendalikan oleh 4 hal, yaitu:

- a. *Neighbor discovery and recovery* : Secara periodic mengirim *hello messages*.
- b. *Reliable Transport Packet (RTP)* : Controls sending, tracking, dan acknowledging EIGRP messages.
- c. *Diffusing Update Algorithm (DUAL)* : Menentukan rute terbaik yang *loop-free*.
- d. *Protocol-independent modules (PDM)* : Modules berupa *plugin*

untuk IP, IPX, dan AppleTalk.

2.2.6 Border Gateway Protocol (BGP)

Border Gateway Protocol atau yang sering disingkat BGP merupakan salah satu jenis *routing* protokol dinamis yang ada di dunia komunikasi data. BGP memiliki kemampuan melakukan pengumpulan informasi rute, pertukaran rute dan menentukan rute terbaik untuk menuju *destination* dalam suatu jaringan. BGP termasuk dalam kategori protokol *routing* jenis *exterior gateway protocol* (EGP), dengan demikian protokol BGP memiliki kemampuan melakukan pertukaran rute dari dan keluar jaringan lokal sebuah organisasi atau kelompok tertentu. Organisasi atau kelompok tertentu diluar organisasi pribadi sering disebut dengan *autonomous system* (AS). Rute – rute yang dimiliki oleh sebuah AS dapat dimiliki oleh AS lainnya yang berbeda kepentingan dan otoritas. (Darmawana dan Teguh Imantob, 2017) .

Menurut Musril (2017), BGP memiliki atribut – atribut pendukung yang menjadi ciri khas dan kekuatan protokol BGP itu sendiri. Atribut- atribut yang nantinya digunakan sebagai parameter untuk menentukan jalur terbaik untuk menuju ke suatu situs. Atribut ini juga dapat mengatur masuk dan keluarnya *routing update* dari *router - router* BGP tetangga. Sedangkan menurut Athira, dkk (2017), BGP adalah protokol yang bertanggung jawab mengelola *network Reachability* (NRI) antar *autonomous system* dengan jaminan menghindari *routing loop*.

Routing protokol BGP baru dapat dikatakan bekerja pada sebuah *router* jika sudah terbentuk sesi komunikasi dengan *router* tetangganya yang juga

menjalankan BGP. Sesi komunikasi ini adalah berupa komunikasi dengan protokol TCP dengan nomor *port* 179. Setelah terjalin komunikasi ini, maka kedua buah *router* BGP dapat saling bertukar informasi rute. Setelah semuanya berjalan dengan baik, maka sebuah sesi BGP dapat bekerja dengan baik pada *router*. BGP merupakan *routing* protokol yang paling besar dan lambat dalam memproses *update routing* informasinya. Hal ini dikarenakan proses *update information* BGP terjadi antara beda *autonomos system* (AS).

Beberapa atribut pada protokol BGP, diantaranya :

- *Aggregator*, yaitu ID dan AS dari *router* yang melakukan summarisasi.
- *AS Path*, yaitu *list autonomous system* yang akan di *advertise*.
- *Atomic Aggregate*, yaitu termasuk kedalam AS-AS yang akan di *drop* karena untuk *route aggregation*.
- *Cluster ID* , yaitu Asal *Cluster*.
- *Community*, yaitu *Route Tag*.
- *Local Preference*, *metric* untuk internal *neighbors* yang digunakan untuk mencapai internal *destination* (*Default* 100).
- *Multiple Exit Discriminator* (MED), *metric* untuk eksternal *neighbors* untuk mencapai eksternal *destination* (*Default* 100).
- *Next Hop* yaitu *peer* eksternal dalam *neighbor* AS.
- *Origin*, yaitu tipe *router* dasarnya (IGP,EGP,atau tidak diketahui dari asalnya).
- *Weight*, yaitu Cisco *Proprietary*, tidak dikomunikasikan dengan *peer* (*default* nya 0).

BGP juga memiliki tipe atribut, diantaranya :

- *Well-known Mandatory*, yaitu harus di ada di semua BGP *router*, ada di semua BGP *update* dan dapat melewati semua BGP *router*. Contoh, AS *path*, *origin* dan *next hop*.
- *Well-knowndiscretionary*, artinya akan ada kalau di konfigurasi : *local preference*.
- *Optional Transitive*, artinya kalau atribut itu tidak di kenali maka atributnya akan dilewatkan. contoh : *aggregator*, *community*.
- *Optional Non-transitive*, artinya kalau atribut itu tidak dikenali maka atribut itu akan di *drop*. Contoh : *Multi-exit Discriminator (MED)*, *Originator ID*.

Protokol BGP memiliki *database* yang mendukung kinerjanya, adapun *database* tersebut diantaranya, :

- *Neighbour Database*, artinya semua *list* yang dikonfigurasi BGP *neighbours*. Perintah yang digunakan pada CLI `show ip bgp summary`.
- BGP *database* atau *routing Information Base (RIB)*, yaitu daftar jaringan yang dikenal dengan BGP, bersama dengan mereka *path* dan atribut. Untuk melihatnya, gunakan perintah pada CLI — `show ip bgp l`.
- BGP *Routing tabel*, yaitu daftar jalan untuk setiap jaringan yang digunakan oleh *router* dan hop berikutnya untuk setiap jaringan. untuk melihat, gunakan perintah `-show ip route bgp`.

Tipe pesan dalam protokol BGP diantaranya, :

- *Open*, yaitu setelah sebuah *neighbour* di konfigurasi, BGP mengirim

open message untuk mencoba *establish peering* antara tetangga itu. termasuk informasi mengenai *autonomous number* (AS), *router ID* dan *hold time*.

- *Update*, yaitu pesan yang digunakan untuk mengirim informasi *routing* antara *peer*. termasuk *route* baru, *route* yang ditarik dan atribut *path*.
- *Keep alive*, yaitu BGP *peer* saling bertukar informasi setiap 60 detik secara *default*. Ini yang membuat *peering session* tetap aktif.
- *Notification*, yaitu ketika sebuah masalah terjadi yang menyebabkan sebuah *router* di *peering* BGP terputus, maka notifikasi dalam bentuk pesan akan dikirim ke *neighbour* dan menyebabkan koneksi itu terputus.

Dalam BGP, ada juga proses penentuan jalurnya, proses penentuan jalur ini nantinya yang akan mempengaruhi konfigurasi yang dibuat. Urut - urutannya adalah sebagai berikut :

- *Weight* (tertinggi) : berdasarkan *administrative preference*.
- *Local Preference* (tertinggi) : berkomunikasi antara *peer* dalam satu AS.
- *Self / Locally Originated* (*true / false*) : Lebih memilih *origin* lokal.
- *AS Path* (terendah) : Memilih AS *hops* yang terendah.
- *Origin* (IGP > EGP > ?) : lebih memilih IGP yang dipelajari *router* lewat EGP.
- *MED* (terendah) : digunakan jalur eksternal untuk masuk kedalam AS.
- *External* (eBGP) : lebih dipilih *routing* yang didapat dari eBGP daripada iBGP.

- IGP *Cost* (terendah) : lebih memilih IGP *metric*.
- eBGP *Peering* (*Oldest*) : lebih menghasilkan *routing* yang stabil.
- *Router ID* (*Lowest*) : dipilih sebagai penentu terakhir dalam penentuan *path*.

Neighbour State / pesan dalam BGP ada 6, diantaranya :

- *Idle* : *neighbour* tidak merespon.
- *Active* : mencoba untuk terhubung.
- *Connect* : TCP *session established*.
- *Open Sent* : *open message* dikirim.
- *Open Confirm* : respon diterima.
- *Establish* : BGP terhubung dengan tetangga.

2.2.7 Redistribution

Redistribution yang didalam Kamus Besar Bahasa Indonesia artinya adalah penyaluran kembali. Kata *redistribution* diserap kedalam bahasa Indonesia redistribusi. Dalam dunia jaringan komputer, terdapat juga istilah ini khususnya dalam peroutingan jaringan komputer. Dalam peroutingan jaringan komputer, *redistribution* merupakan sebuah metode yang digunakan untuk meneruskan suatu *routing* protokol ke *routing* protokol lain yang berbeda agar dapat saling berbagi informasi *routing* tabel masing – masing atau cara untuk meredistribusikan kembali *routing* tabel yang dibentuk oleh suatu *routing* protokol untuk diteruskan ke *routing* protokol lain yang berbeda.

Menurut Maulana (2018), *redistribution* adalah metode *routing* protokol yang digunakan untuk meneruskan suatu *routing* protokol ke *routing* protokol yang lain agar dapat saling menukarkan *routing* tabel masing-masing. Dengan metode *redistribution* ini, *routing* yang memiliki matrik dan pemilihan jalur pengiriman yang berbeda dapat mengirimkan data satu sama lainnya. Hal ini akan membuat lebih mudah bagi *network administrator* untuk mengelola jaringan komputer yang dibutuhkan. Teknik *redistribution* juga dapat diartikan sebagai solusi untuk melakukan komunikasi antar *routing* protokol (Aryanta & Pranata, 2014).

2.2.8 Transmission Control Protocol / Internet Protocol (TCP/ IP)

TCP/IP merupakan protokol yang digunakan dalam jaringan komputer dan menjadi protokol standar yang digunakan internet. Internet protokol yang banyak digunakan sampai saat ini adalah internet protokol versi 4 atau yang dikenal dengan ipv4. (Iwan, 2012).

Menurut Kozierok (2005), TCP/IP merupakan sebuah aturan standar komunikasi data yang berfungsi dalam memproses pengiriman data dari satu komputer ke komputer lain dalam suatu jaringan komputer meskipun jenis hardware berbeda. Sedangkan menurut Forouzan dan Chung Fegan (2007, pp 43 - 46), TCP/IP merupakan sebuah tingkatan protokol berupa modul – modul interaktif, dimana setiap modul tersebut memiliki fungsionalitas yang sangat spesifik.

A. Format IP

Setiap alamat IP memiliki format yang berbeda-beda yang terdiri dari sebuah bagian host dan sebuah bagian jaringan. Di dalam alamat IP terdapat beberapa bit alamat yang dipakai untuk mengenali jaringan, sedangkan angka dipakai untuk mengenali host serta berbagai kelas alamat IP. Alamat IP memiliki 5 kelas utama, yakni kelas A, B, C, D dan E dengan ketentuan sebagai berikut:

1. Jaringan Kelas A

Jaringan kelas A ditandai dengan adanya angka 0 pada bit pertama alamat IP. Kemudian tujuh bit setelahnya menunjukkan identitas jaringan dan 24 bit terakhir menunjukkan identitas *host*. Pada jaringan kelas A terdapat 128 angka jaringan dan jutaan *host*. Berikut beberapa klasifikasi ip address kelas A, diantaranya :

- Pemakaian: Banyak digunakan dalam jaringan komputer berskala besar.
- Jumlah jaringan: 126 serta masing – masing jaringan mampu menampung sebanyak 16,777,214 host.
- *Range* angka: 0.0.0.0 – 127.255.255.255

2. Jaringan Kelas B

Jaringan kelas B ditandai dengan dua bit pertama dari alamat ip berupa angka 10. Angka bit pertama merupakan kelas, 24 bit setelahnya menunjukkan identitas alamat jaringan, dan 10 bit setelahnya adalah *host*. Pada jaringan kelas A terdapat ribuan angka jaringan dan ribuan *host*. Berikut beberapa klasifikasi ip address kelas B, diantaranya :

- Pemakaian: Jaringan komputer skala kecil.
- Jumlah maksimal jaringan: 16.384.

- Jumlah maksimal alamat IP: 1.048.576.
- *Range* angka: 128.0.0.0 – 191.255.255.255.

3. Jaringan Kelas C

Jaringan kelas C ditandai dengan tiga bit pertama dari alamat ip berupa angka 110. Tiga bit pertama tersebut berupa alamat kelas, 21 bit setelahnya menunjukkan alamat jaringan, dan 8 bit setelahnya adalah identitas *host*. Pada jaringan kelas C terdapat jutaan angka jaringan dan 254 *host*. . Berikut beberapa klasifikasi ip address kelas C, diantaranya :

- Pemakaian: Jaringan komputer skala kecil.
- Jumlah maksimal jaringan: 2.097.152.
- Jumlah maksimal alamat IP: 65.536.
- *Range* angka: 192.0.0.0 – 223.255.255.255.

4. Jaringan Kelas D

IP Address kelas D digunakan untuk keperluan *multicasting*. 4 bit pertama pada IP address kelas D selalu diatur 1110, sehingga *byte* pertamanya berkisar antara 224-247 dalam bilangan desimal. Sedangkan bit-bit selanjutnya diatur sesuai dengan keperluan *multicast group* yang menggunakan IP address ini. Dalam *multicasting*, tidak dikenal istilah *network ID* dan *host ID*. Berikut beberapa klasifikasi ip address kelas D, diantaranya :

- Pemakaian: IP *multicas*.
- Jumlah maksimal jaringan: tidak terdefinisi.
- Jumlah maksimal alamat IP: tidak terdefinisi.
- *Range* angka: 224.0.0.0 – 239.255.255.255.

5. Jaringan Kelas E

IP address kelas E sebenarnya tidak diperuntukkan untuk keperluan umum. 4 bit pertama pada IP address kelas ini selalu diatur 1111, sehingga *byte* pertamanya berkisar antara 248-255 dalam bilangan desimal.

Berikut beberapa klasifikasi ip address kelas E, diantaranya :

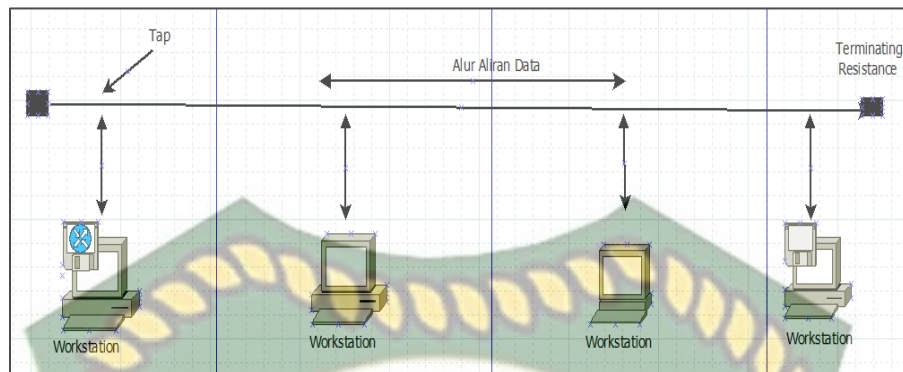
- Pemakaian: Eksperimen.
- Jumlah maksimal jaringan: tidak terdefinisi.
- Jumlah maksimal alamat IP: tidak terdefinisi.
- *Range* angka: 140.0.0.0 – 255.255.255.255.

2.2.9 Topologi Jaringan

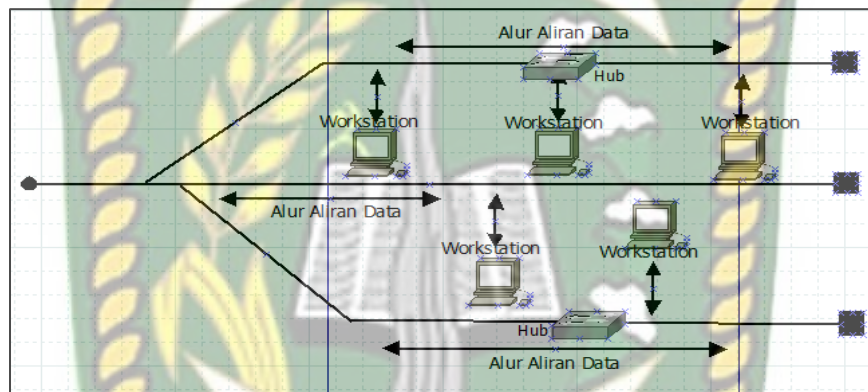
Menurut Sofana (2013:7), topologi dapat diartikan sebagai *layout* atau arsitektur atau diagram jaringan komputer. Topologi merupakan suatu aturan / rules bagaimana menghubungkan komputer secara fisik. Topologi berkaitan dengan cara komponen-komponen jaringan seperti: *server*, *workstation*, *switch* untuk saling berkomunikasi melalui media transmisi data. Topologi jaringan pada umumnya terdapat beberapa yaitu :

A. Topologi *Bus* dan *Tree*

Topologi *Bus* dan *tree* menggunakan media multipoin. Untuk topologi *bus* semua stasiun terjangkau, melalui satu *interface* yang terhubung terus dan di ujung cabangnya terdapat terminal. Untuk topologi *tree* atau pohon yaitu bentuk lain dari topologi *bus* yaitu dengan cabang kabel yang bukan *loop* tertutup. Bentuk topologi *bus* dapat dilihat pada gambar 2.1 dan topologi *tree* pada gambar 2.2.



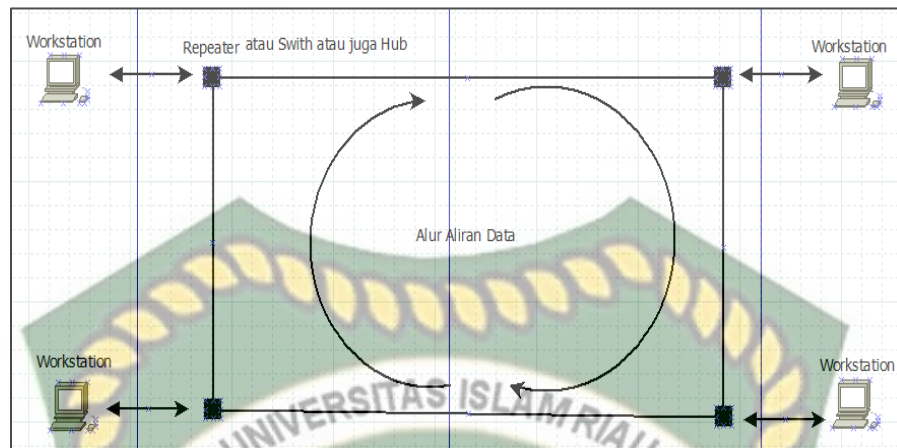
Gambar 2.1 Topologi Bus.



Gambar 2.2 Topologi Tree.

B. Topologi Ring

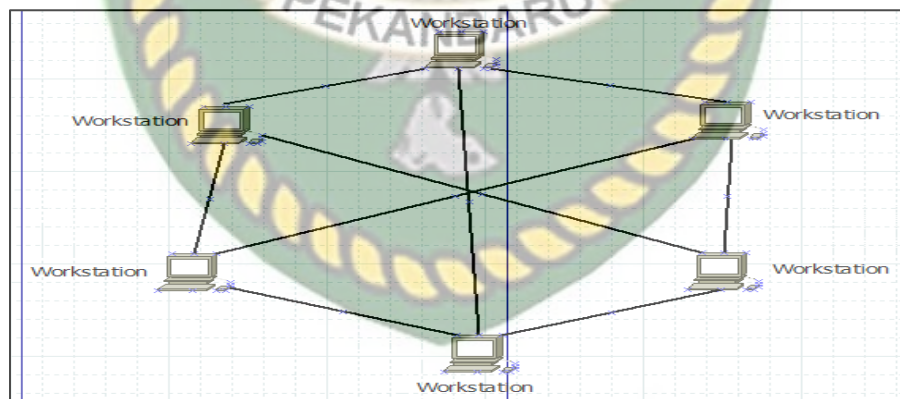
Topologi ini aliran *flow* data *point to point* berbentuk *ring link* bersifat *loop* tertutup, karena *link* yang bersifat *unidireksional*, artinya data hanya diterima hanya satu arah, sehingga data hanya disirkulasi satu arah saja. Sehingga kecepatan data sama dengan dengan yang lainnya yang masih dalam satu *ring*. Bentuk topologi *ring* dapat dilihat pada gambar 2.3 berikut.



Gambar 2. 3 Topologi *Ring*.

C. Topologi *Mesh*

Pada topologi ini setiap komputer akan terhubung dengan komputer lain dalam jaringannya menggunakan kabel tunggal, jadi proses pengiriman data akan langsung mencapai komputer tujuan tanpa melalui komputer lain ataupun *switch* atau *hub*. Bentuk topologi *mesh* dapat dilihat pada gambar 2.4 berikut.

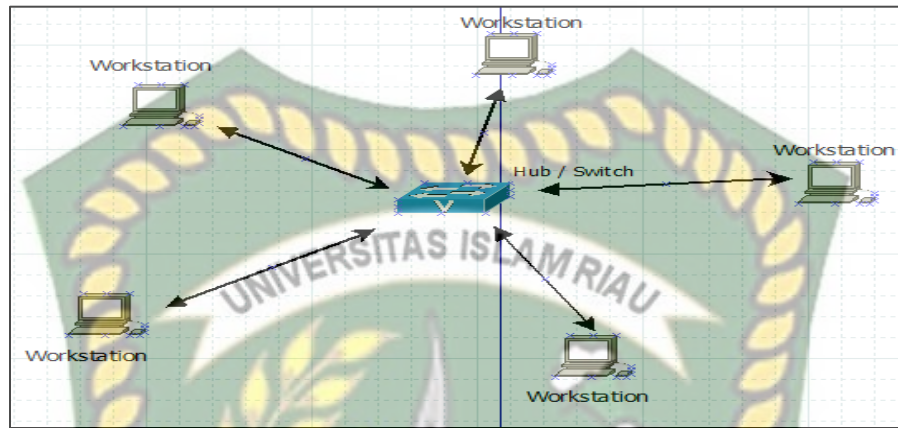


Gambar 2. 4 Topologi *Mesh*.

D. Topologi *Star*

Dalam topologi LAN *star* atau dalam bahasa Indonesia disebut dengan bintang, setiap stasiun terhubung langsung ke *node* pusat. Biasanya, setiap stasiun

menempel ke *node* pusat melalui dua *point-to-point*, satu untuk transmisi dan satu untuk penerima. Bentuk topologi *star* dapat dilihat pada gambar 2.5 berikut.









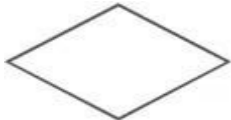

Gambar 2. 5 Topologi *Star*.


2.2.10 *Flowchart*

Flowchart adalah bagan - bagan yang mempunyai arus yang menggambarkan langkah - langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma. Simbol *flowchart* dan fungsinya dapat dilihat pada tabel 2.1 berikut :

Tabel 2. 1 Simbol dan Keterangan Fungsi *Flowchart*.

| No | Simbol | Nama | Fungsi |
|----|---|-------------------|---------------------------------|
| 1 |  | <i>Terminator</i> | Permulaan / pengakhiran program |
| 2 |  | <i>Flow Line</i> | Arah aliran program |

| | | | |
|---|---|---------------------------|--|
| 3 |  | <i>Preparation</i> | Proses inisialisasi/pemberian nilai awal |
| 4 |  | <i>Process</i> | Proses pengolahan data |
| 5 |  | <i>Input/Output Data</i> | Proses input/output data, parameter, informasi |
| 6 |  | <i>Predefined Process</i> | Permulaan sub program / proses menjalankan sub program |
| 7 |  | <i>Decision</i> | Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya |
| 8 |  | <i>On Page Connector</i> | Penghubung bagian-bagian flowchart yang berada pada satu halaman |

| | | | |
|---|---|---------------------------|--|
| 9 |  | <i>Off page Connector</i> | Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda |
|---|---|---------------------------|--|

2.3 Alat dan Bahan Penelitian

2.3.1 Perangkat Keras

Perangkat keras yang digunakan pada penelitian ini adalah sebuah laptop, dengan spesifikasi sebagai berikut :

1. Laptop Acer E5-471-34VN Sistem Operasi Windows 10, Processor Intel(R)Core(TM)i3-6006U CPU 2.0Ghz, RAM Laptop 4.00 GB, 500 GB HDD, Sistem tipe 64 bit *operating System, x64-based processor.*
2. Laptop Acer One 14 Z476 Sistem Operasi Windows 10, Processor Intel(R)Core(TM)i3-6006U CPU 2.0Ghz, RAM Laptop 4.00 GB, 1TB, Sistem tipe 64 bit *operating System, x64-based processor.*
3. *Router cisco 2900 dan 2901.*

2.3.2 Perangkat Lunak

Perangkat lunak yang digunakan pada penelitian ini adalah sebagai berikut :

1. Windows Server 2008 (64 bit) (SO virtual dalam Virtual Box).
2. Sistem Operasi Windows 10).
3. Putty (*putty - 64bit - 0.76-installer*).
4. Wireshark *version 3.4.7 64-bit.*
5. Oracle Virtual Box *Version 6.1.22 r144080 (Qt5.6.2)*

2.3.3 Bahan Penelitian

Dalam melakukan penelitian ini, penulis menggunakan beberapa jurnal penelitian yang telah dijabarkan secara garis besar penulis pada bab 2 karya tulis ini, sebagai bahan acuan untuk penelitian.

2.3.4 Aplikasi PuTTY

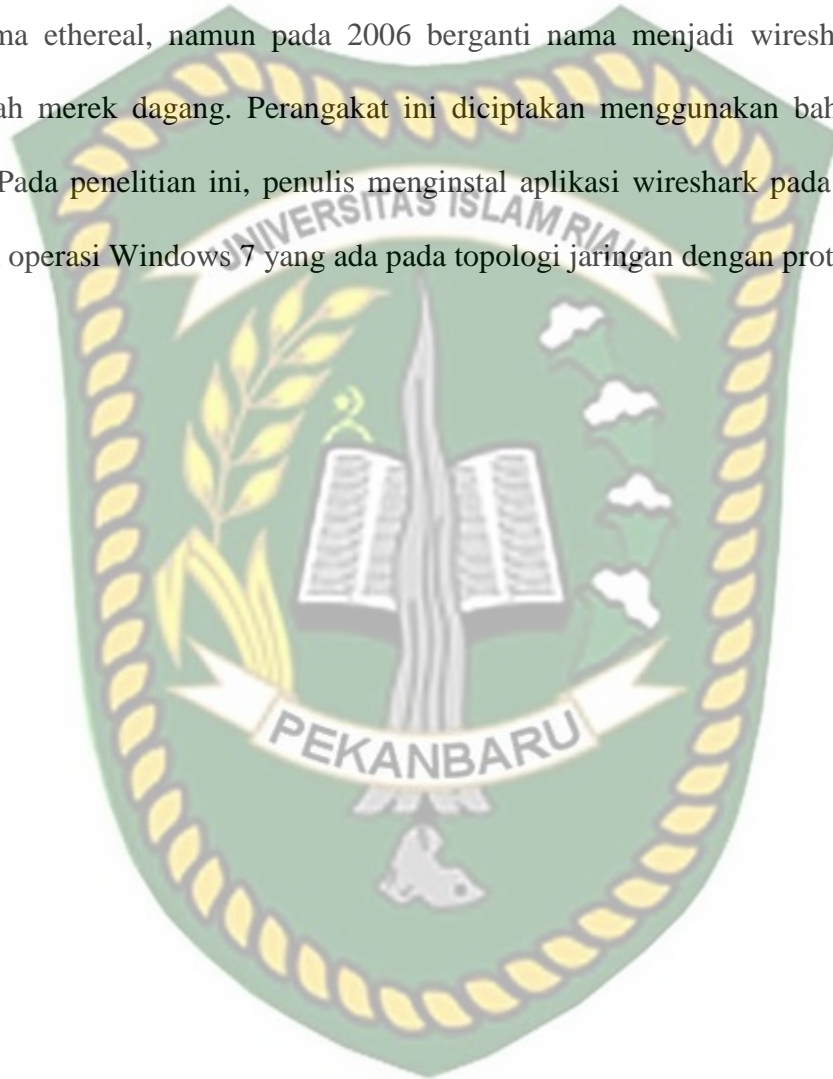
PuTTY merupakan sebuah aplikasi *open-source* yang mendukung banyak protokol seperti SSH, Telnet, SCP, rlogin, *serial port*, dan *raw socket connection*. Komunikasi Putty dibuat melalui TCP / IP socket, seperti TTY tetapi menggunakan secure socket dengan enkripsi *public key*. Perintah yang diketik adalah dikirim dan menerima respon teks.

PuTTY memiliki beberapa fitur penting:

- PuTTY mendukung banyak variasi pada *secure remote* terminal.
- Memberikan kontrol atas SSH *encryption key* dan versi protokol.
- Menyimpan *host* dan preferensi untuk penggunaan nanti.
- Baris perintah SCP dan klien SFTP.
- Meniru urutan kontrol dari *xterm*, VT102 atau emulasi terminal ECMA-48.
- Memungkinkan remote, lokal atau *dynamic port forwarding* dengan SSH.
- Digunakan dengan *local serial port connections*.

2.3.5 Wireshark

Wireshark merupakan perangkat lunak yang digunakan untuk menganalisa paket dan *open-soure* pada suatu jaringan komputer. Perangkat lunak ini awalnya bernama ethereal, namun pada 2006 berganti nama menjadi wireshark karena masalah merek dagang. Perangkat ini diciptakan menggunakan bahasa C dan C++. Pada penelitian ini, penulis menginstal aplikasi wireshark pada pc dengan sistem operasi Windows 7 yang ada pada topologi jaringan dengan protokol BGP.



BAB III

METODE PENELITIAN

3.1 Metode Penelitian

Metode penelitian merupakan sebuah alur kerja yang dilakukan oleh seorang peneliti dengan mengumpulkan data atau informasi yang berhubungan dengan penelitian yang dilakukan dan melakukan analisa terhadap data atau informasi tersebut untuk menghasilkan kesimpulan dari penelitian yang dilakukan. Metode penelitian memiliki beberapa pengertian berdasarkan pendapat ahli diantaranya, :

1. Muhidin sirat, menjelaskan bahwa metode penelitian merupakan sebuah cara berupa langkah - langkah memilih masalah dan penentuan judul.
2. Winarno, menjelaskan bahwa metode penelitian merupakan suatu kegiatan yang dilakukan dengan teliti dan sistematis.
3. Nasir, menjelaskan bahwa metode penelitian merupakan sebuah cara yang dilakukan peneliti untuk mencapai tujuan penelitian yang dilakukan serta menentukan jawaban atas permasalahan yang diajukan.

Pada penelitian ini penulis membagi metode penelitian menjadi 4 tahap, yaitu :

1. Tahap pengumpulan data.
2. Tahap perancangan.
3. Tahap konfigurasi.
4. Tahap pengujian.

1. Tahap Pengumpulan Data

Dalam melakukan pengumpulan data, penulis menganalisa data yang terdapat dalam jurnal yang dijadikan rujukan, mempelajari data tersebut untuk dijadikan acuan dalam membuat sebuah simulasi topologi jaringan dengan menggunakan perangkat nyata, pembagian ip dalam jaringan dan bagaimana memproses paket – paket yang menjadi nilai dalam analisa.

2. Tahap Perancangan

Tahap perancangan merupakan tahapan yang peneliti lakukan dalam merancang topologi jaringan komputer untuk melakukan analisa kinerja *redistribution* protokol OSPF, EIGRP dan BGP. Pada penelitian ini perancangan topologi jaringan menggunakan perangkat nyata. Perangkat tersebut diantaranya terdiri dari 3 buah *router* cisco dengan rincian 2 *router* cisco 2900 dan 1 buah *router* cisco 2901. Selain itu untuk perangkat akhir digunakan 2 buah laptop Acer dengan sistem operasi windows 10 dengan rincian 1 laptop sebagai *client* dan 1 laptop lagi sebagai server dengan menggunakan virtualbox untuk menjalankan Windows Server 2008.

Topologi jaringan yang dirancang diasumsikan memiliki area yang terhubung dengan *routing* OSPF, EIGRP dan BGP. Perangkat yang digunakan pada topologi jaringan ini menggunakan perangkat jaringan cisco.

3. Tahap Konfigurasi

Pada Tahap ini, peneliti melakukan konfigurasi perangkat jaringan. Pada setiap *router* peneliti melakukan konfigurasi *routing* protokol baik itu OSPF, EIGRP dan BGP dan juga memberikan pengalamatan ip pada *router* sesuai area

yang telah ditetapkan ip jaringannya dan juga memberikan pengalamatan ip pada PC.

4. Tahap Pengujian

Tahap pengujian dilakukan setelah konfigurasi. Pengujian dilakukan setelah seluruh perangkat keras dapat terhubung dan berkomunikasi dengan baik. Dalam proses pengujian ini, peneliti melakukan analisa terhadap nilai *delay*, *throughput* dan *packet loss* pada paket data yang dikirim menggunakan layanan ICMP.

Untuk mendapatkan nilai dari 3 parameter tersebut, peneliti menggunakan perangkat lunak Wireshark yaitu sebuah perangkat lunak yang digunakan menganalisa sebuah jaringan komputer. Berikut merupakan penjelasan singkat tentang parameter yang penulis gunakan dalam penelitian ini :

A. *Throughput*

Throughput merupakan kecepatan rata-rata yang diterima oleh sebuah perangkat keras dalam sebuah jaringan ketika antar perangkat keras tersebut berkomunikasi dalam waktu pengamatan tertentu. *Throughput* merupakan *bandwidth* aktual saat itu juga dimana kita sedang melakukan koneksi. Satuan yang dimilikinya sama dengan *bandwidth* yaitu bps (*byte per second*).

Secara matematis *Throughput* dapat dihitung dengan rumus :

$$\textit{Throughput} = \frac{\text{Jumlah data yang dikirim}}{\text{Waktu pengiriman data}} \quad (3.1)$$

Selain dengan perhitungan diatas, nilai *throughput* dapat dicari dengan menggunakan *software* wireshark. Berikut contoh nilai *throughput* pada wireshark.



Gambar 3. 1 Pencarian Nilai *Throughput* pada Wireshark.

pada gambar diatas terlihat nilai dari *throughput* yang didapatkan adalah 0.272 MBit/second.

B. Packet Loss

Packet Loss adalah banyaknya paket yang hilang pada suatu jaringan paket yang disebabkan oleh tabrakan (*collision*), penuhnya kapasitas jaringan, dan penurunan paket yang disebabkan oleh habisnya TTL (Time To Live) paket.

Secara matematis *packet loss* dapat dihitung dengan rumus :

$$packet\ loss = \left(\frac{\text{paket data yg dikirim} - \text{paket data diterima}}{\text{paket data dikirim}} \right) \times 100\%. \quad (3.2)$$

Selain dengan perhitungan diatas, nilai *packet loss* dapat dicari dengan menggunakan *software* wireshark. Gambar 3.2 berikut menampilkan contoh pencarian nilai *packet loss* pada wireshark.

Wireshark: RTP Stream Analysis

Forward Direction: Reversed Direction

Analyzing stream from 192.168.7.21 port 33442 to 192.168.7.100 port 11670 SRC = 2010190AE7

| Packet | Sequence | Delta(ms) | Fibered jitter(ms) | Skew(ms) | IP Diffs (ms) | Marker | Status |
|--------|----------|-----------|--------------------|----------|---------------|--------|--------------------|
| 245 | 2216 | 0.00 | 0.00 | 0.00 | 1.00 | SET | [OK] |
| 252 | 2217 | 12.71 | 0.46 | 2.29 | 3.20 | | [OK] |
| 254 | 2218 | 14.55 | 0.77 | 12.24 | 4.80 | | [OK] |
| 258 | 2219 | 19.71 | 0.74 | 13.03 | 6.40 | | [OK] |
| 262 | 2220 | 21.61 | 0.78 | 11.41 | 8.00 | | [OK] |
| 270 | 2222 | 33.84 | 0.75 | 11.58 | 8.00 | | Wrong sequence no. |
| 274 | 2223 | 13.49 | 0.80 | 13.03 | 11.20 | | [OK] |
| 278 | 2224 | 20.85 | 0.78 | 13.03 | 12.80 | | [OK] |

Max delta = 104.06 ms at packet no. 9140
 Max jitter = 17.04 ms, Mean jitter = 4.07 ms
 Max skew = -147.92 ms
 Total RTP packets = 3101 (expected 3100) Lost RTP packets = 45 (1.45%) Sequence errors = 45
 Duration 62.43 s [-30 ms clock drift, corresponding to 7995 Hz (-0.26%)

Save payload... Save as CSV... Refresh Jump to Graph Player Next non-Ok Close

Gambar 3. 2 Pencarian Nilai *Packet Loss* pada Wireshark.

Dari gambar diatas nilai dari *packet loss* yang didapatkan adalah 1.45 %.

C. Delay

Delay adalah waktu tunda saat paket yang diakibatkan oleh proses transmisi dari satu titik menuju titik lain yang menjadi tujuannya.

Secara matematis nilai *delay* dapat dihitung dengan rumus :

$$\text{Rata - rata delay} = \frac{\text{total delay}}{\text{total paket yang diterima}} \quad (3.3).$$

Selain dengan perhitungan diatas, nilai *delay* dapat dicari dengan menggunakan *software* wireshark. Gambar 3.3 berikut menampilkan contoh pencarian nilai *delay* pada wireshark.

Wireshark: RTP Stream Analysis

Forward Direction: Reversed Direction

Analyzing stream from 192.168.7.21 port 33442 to 192.168.7.100 port 11670 SRC = 2010190AE7

| Packet | Sequence | Delta(ms) | Fibered jitter(ms) | Skew(ms) | IP Diffs (ms) | Marker | Status |
|--------|----------|-----------|--------------------|----------|---------------|--------|--------------------|
| 245 | 2216 | 0.00 | 0.00 | 0.00 | 1.00 | SET | [OK] |
| 252 | 2217 | 12.71 | 0.46 | 2.29 | 3.20 | | [OK] |
| 254 | 2218 | 14.55 | 0.77 | 12.24 | 4.80 | | [OK] |
| 258 | 2219 | 19.71 | 0.74 | 13.03 | 6.40 | | [OK] |
| 262 | 2220 | 21.61 | 0.78 | 11.41 | 8.00 | | [OK] |
| 270 | 2222 | 33.84 | 0.75 | 11.58 | 8.00 | | Wrong sequence no. |
| 274 | 2223 | 13.49 | 0.80 | 13.03 | 11.20 | | [OK] |
| 278 | 2224 | 20.85 | 0.78 | 13.03 | 12.80 | | [OK] |

Max delta = 104.06 ms at packet no. 9140
 Max jitter = 17.04 ms, Mean jitter = 4.07 ms
 Max skew = -147.92 ms
 Total RTP packets = 3101 (expected 3100) Lost RTP packets = 45 (1.45%) Sequence errors = 45
 Duration 62.43 s [-30 ms clock drift, corresponding to 7995 Hz (-0.26%)

Save payload... Save as CSV... Refresh Jump to Graph Player Next non-Ok Close

Gambar 3. 3 Pencarian Nilai *Delay* pada Wireshark.

Pada gambar 3.3 diatas, nilai total delaynya 62.03 *second*, dan total paket yang diterima adalah 3103, sehingga jika hitung sesuai dengan rumus didapatkan :

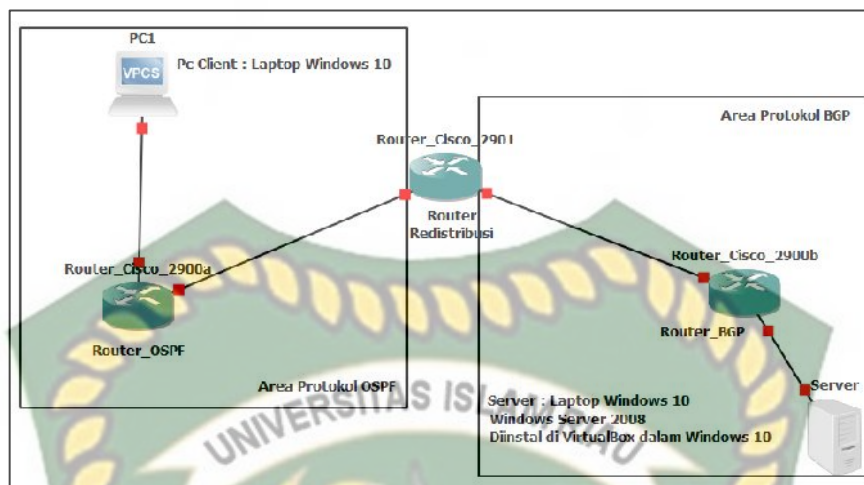
Rata Rata Delay = $62.03 / 3103 = 0.019990$ *second*.

3.2 Desain Topologi

Dalam mendesain topologi jaringan dalam bentuk simulai pada penelitian ini, penulis menggunakan 2 laptop dengan sistem operasi Windows 10 dengan rincian satu digunakan sebagai clien dan 1 lagi digunakan sebagai sever dengan bantuan Virtual Box untuk penginstalan Windows Sever 2008, 3 *router* cisco dengan rincian 2 jenis *router* 2900 dan 1 jenis *router* 2901.

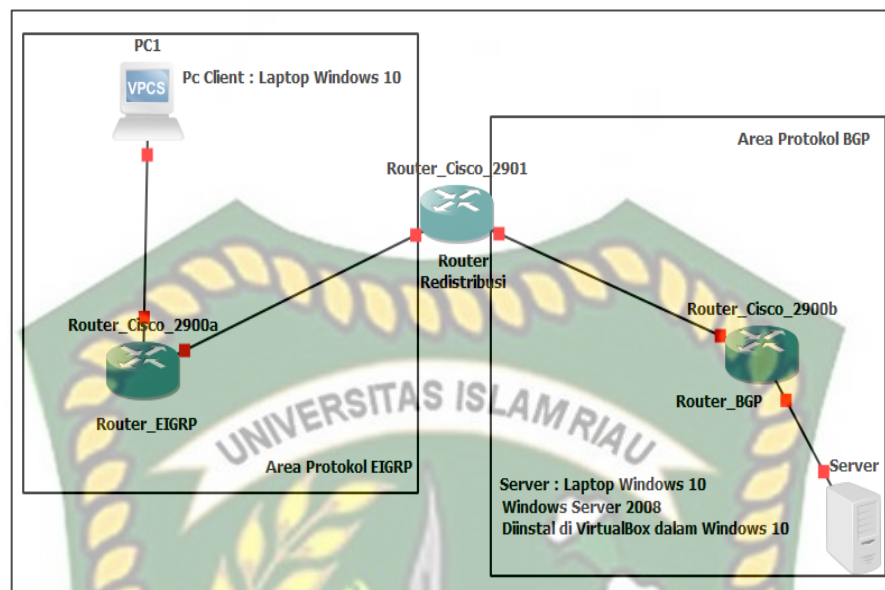
3.2.1 Topologi Fisik

Topologi Fisik merupakan sebuah gambaran topologi secara fisik yang menjelaskan bahwa dalam topologi terdapat bagian - bagian yang membangun dari topologi tersebut, misal berapa buah *device* yang digunakan, pembagian area atau ruangan untuk meletakkan *device – device* yang digunakan . Pada penelitian ini, penulis membagi perancangan topologi fisik menjadi 2, yaitu topologi redistribusi antra BGP dengan OSPF dan redistribusi BGP dengan EIGRP. Dibawah ini merupakan gambar bentuk topologi fisik yang dirancang penulis dalam penelitian yang dilakukan. Dimana terdapat 1 pc (Windows 10), 1 server (Windos server 2008), 3 *router*. Gambar 3.4 berikut menampilkan laptop yang penulis gunakan sebagai clien. Bentuk topologi jaringan redistribusi antra BGP dengan OSPF yang penulis rancang dapat dilihat gambar 3.4dibawah ini.



Gambar 3. 4 Topologi Fisik Redistribusi OSPF dan BGP.

Pada gambar diatas, penulis membuat 2 area protokol yaitu area protokol OSPF dan area protokol BGP. Area protokol OSPF penulis rancang sebagai area bagian topologi clien yang artinya area ini dijadikan sebagai sumber dimulainya proses komunikasi *ping* yang dilakukan dan area protokol BGP sebagai area tujuan atau destinasi. Dari gambar diatas juga menjelaskan bahwa penulis merancang 2 buah *Local Area Network* (LAN) dengan masing – masing protokol OSPF dan BGP . Ditengah kedua LAN pada gambar diatas, terdapat sebuah *router* yang berfungsi sebagai *router* redistribusi. Untuk topologi fisik redistribusi EIGRP dan BGP, dapat dilihat dari gambar 3.5 berikut.



Gambar 3. 5 Topologi Fisik Redistribusi EIGRP dan BGP.

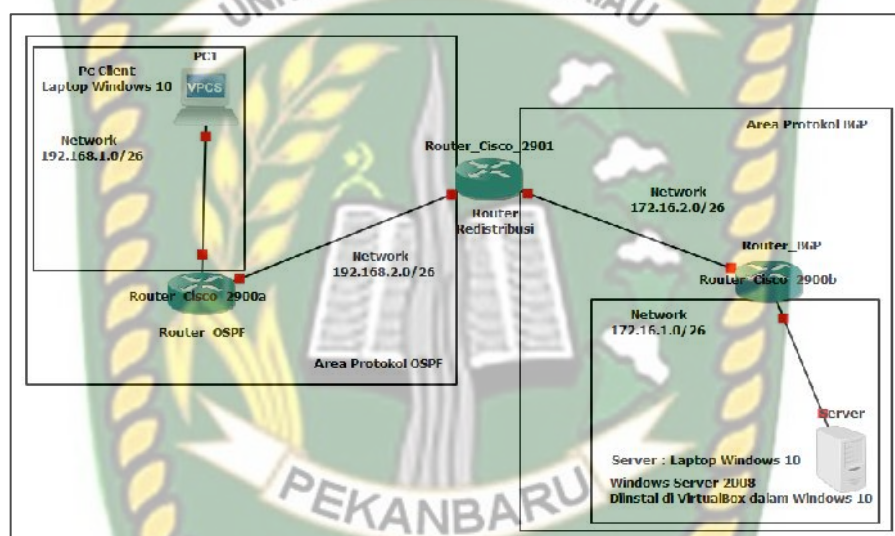
Pada gambar 3.5 diatas, penulis juga membuat 2 area protokol yaitu area protokol EIGRP dan area protokol BGP. Area protokol EIGRP penulis rancang sebagai area bagian topologi clien yang artinya area ini dijadikan sebagai sumber dimulainya proses komunikasi *ping* yang dilakukan dan area protokol BGP sebagai area tujuan atau destinasi. Dari gambar diatas juga menjelaskan bahwa penulis merancang 2 buah *Local Area Network* (LAN) dengan masing – masing protokol OSPF dan BGP . Ditengah kedua LAN pada gambar diatas, terdapat sebuah *router* yang berfungsi sebagai *router* redistribusi.

Tujuan peneliti membuat topologi menjadi 2 bagian terpisah ini adalah karena keterbatasan perangkat *router* cisco yang dimiliki.

3.2.2 Topologi Logical

Sama dengan bentuk topologi fisik, tetapi topologi *logical* menggambarkan secara logika hubungan antar *device* dengan menggunakan ip dalam jaringan. Pada topologi ini, penulis memberikan area mana saja yang dikonfigurasi

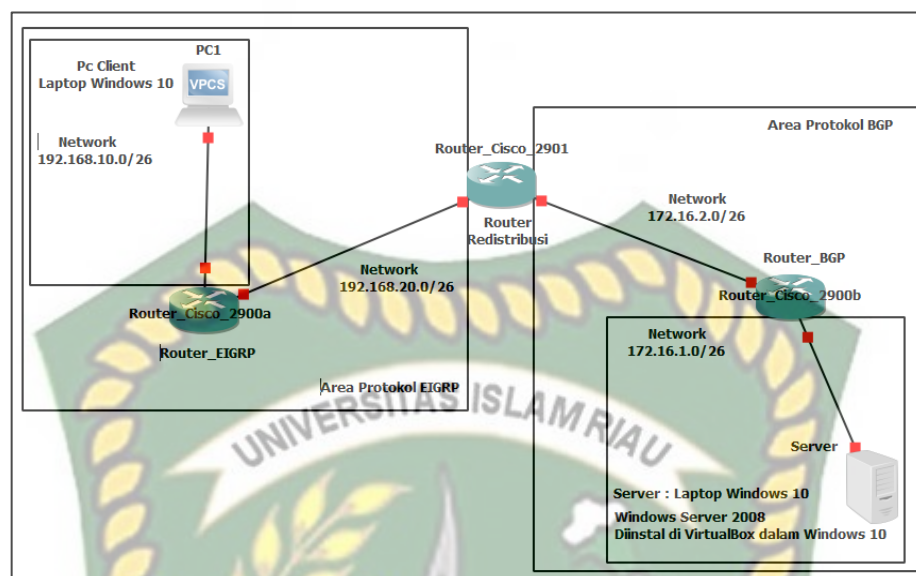
menggunakan protokol OSPF, EIGRP dan BGP. Perancangan topologi *logic* pada penelitian ini juga penulis bagi menjadi 2, yaitu redistribusi OSPF dengan BGP dan redistribusi EIGRP dan BGP hal ini dilakukan karena keterbatasan *router* cisco yang tersedia. Untuk area BGP, proses konfigurasi ip pada topologi redistribusi sama. Gambar 3.6 berikut menampilkan topologi *logic* redistribusi OSPF dan BGP.



Gambar 3. 6 Topologi *Logic* Redistribusi OSPF dan BGP.

Pada redistribusi OSPF dan BGP, area OSPF peneliti memberikan ip *network* 192.168.1.0/26 untuk area penghubung Router_OSPF dengan perangkat akhir (laptop clien) dan ip *network* 192.168.2.0/26 untuk area penghubung Router_OSPF dengan Router_Redistribusi. Sementara untuk area BGP peneliti memberikan IP *Network* 172.16.1.0/26. area penghubung Router_OSPF dengan perangkat akhir (laptop/server) ip *network* 172.16.2.0/26 untuk area penghubung Router_BGP dengan Router_Redistribusi.

Untuk Bentuk topologi redistribusi EIGRP dan BGP, dilihat dari gambar 3.7 berikut.



Gambar 3. 7 Topologi *Logic* Redistribusi EIGRP dan BGP.

Pada redistribusi EIGRP dan BGP, area EIGRP peneliti memberikan ip *network* 192.168.10.0/26 untuk area penghubung Router_EIGRP dengan perangkat akhir (laptop clien) dan ip *network* 192.168.20.0/26 untuk area penghubung Router_EIGRP dengan Router_Redistribusi. Sementara untuk area BGP peneliti memberikan IP *Network* 172.16.1.0/26. area penghubung Router_OSPF dengan perangkat akhir (laptop/server) ip *network* 172.16.2.0/26 untuk area penghubung Router_BGP dengan Router_Redistribusi.

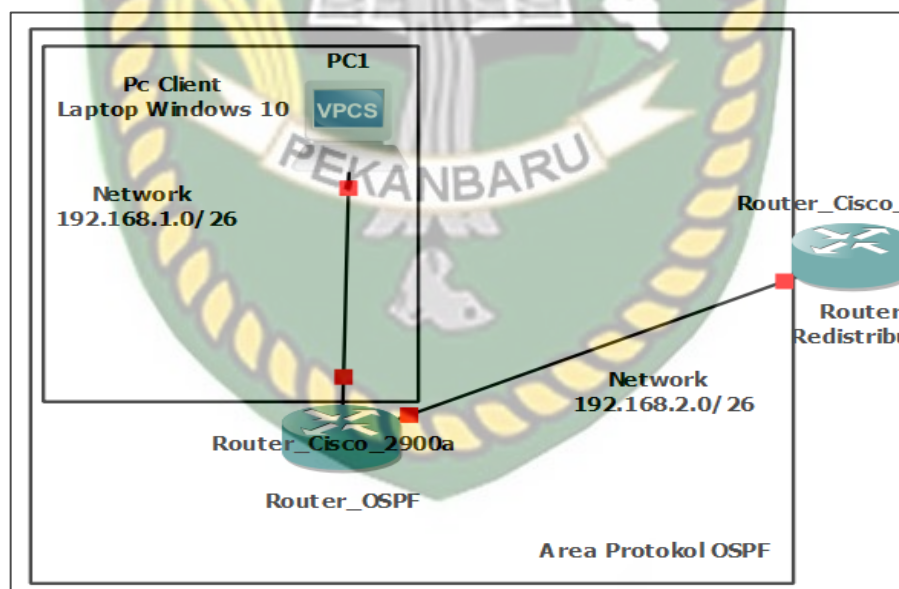
Router redistribusi merupakan *router* yang penulis gunakan untuk melakukan *redistribution routing protocol*. Tujuan dari *redistribution routing protocol* adalah agar jaringan yang memiliki protokol dinamis yang berbeda dapat berkomunikasi. Pada penelitian ini, digunakan 3 protokol dinamis yaitu OSPF, BGP dan EIGRP.

3.3 Perancangan Jaringan Menggunakan Protokol

Pada tahap ini, penulis melakukan perancangan jaringan menggunakan protokol *routing* dinamis. Pada penelitian ini, penulis menggunakan 3 protokol dinamis yaitu OSPF, EIGRP, dan BGP.

3.3.1 Jaringan Protokol OSPF

Pada jaringan OSPF ini, terdapat 1 buah *router* yang penulis namakan Router_OSPF. *Router* ini berfungsi penghubung antara jaringan 192.168.1.0/26 dengan 192.168.2.0/26 dimana dari jaringan 192.168.2.0 inilah yang juga akan menghubungkan jaringan 192.168.1.0 dengan router redistribusi. Gambar 3.8 berikut menggambarkan topologi jaringan OSPF yang penulis rancang.



Gambar 3. 8 Topologi Jaringan Protokol OSPF.

Dari gambar 3.8 diatas terdapat sebuah pc dimana pada perangkat nyata yang penulis gunakan adalah sebuah laptop dengan sistem operasi Windows 10 yang penulis set sebagai PC klien. Gambar 3.9 berikut merupakan tampilan perangkat nyata yang penulis gunakan sebagai perangkat klien.



Gambar 3. 9 Perangkat Nyata Laptop Klien pada OSPF.

Untuk konfigurasi ip pada perangkat klien, penulis melakukan DHCP dengan melakukan konfigurasi pada Router_OSPF. Untuk pengaturan ip *interface* (kabel) pada Router_OSPF dapat dilihat pada gambar 3.10 berikut



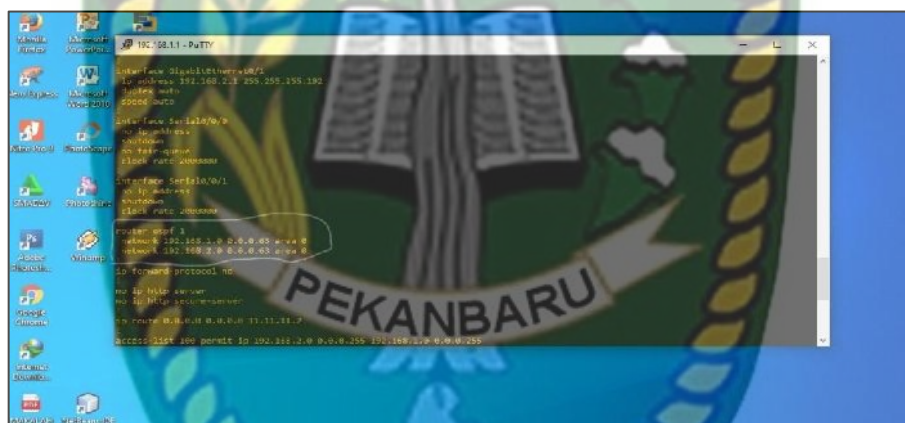
Gambar 3. 10 Konfigurasi IP Interface pada Router_OSPF.

Pada gambar 3.10 diatas, *interface* (slot kabel) yang digunakan pada Router_OSPF adalah *interface* GigabitEthernet0/0 (GE0/0) dengan ip 192.168.1.1 dan *interface* GigabitEthernet0/1 (GE0/1) dengan ip 192.168.2.1. Gambar 3.11 berikut ini menampilkan slot pada Router_OSPF.



Gambar 3. 11 Slot GE0/0 dan GE0/1 pada Router_OSPF.

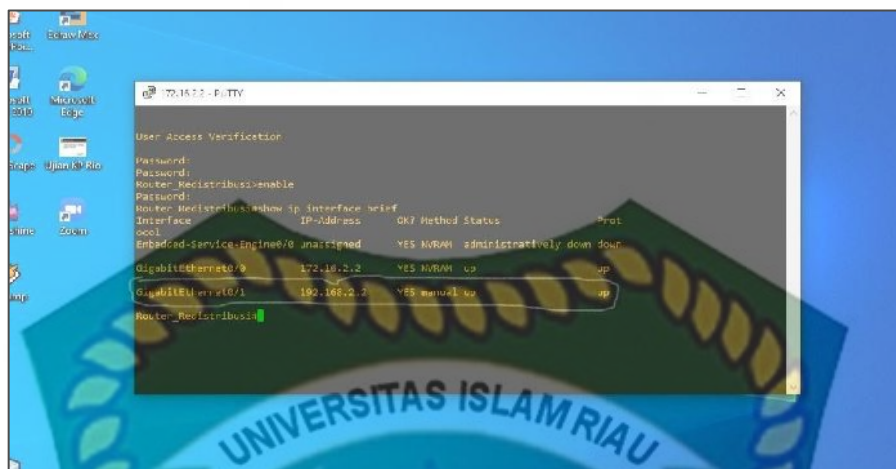
Pada Router_OSPF, penulis melakukan konfigurasi OSPF dengan area 0. Gambar 3.12 berikut ini menampilkan konfigurasi protokol OSPF di Router_OSPF.



Gambar 3. 12 Konfigurasi Protokol OSPF pada Router_OSPF.

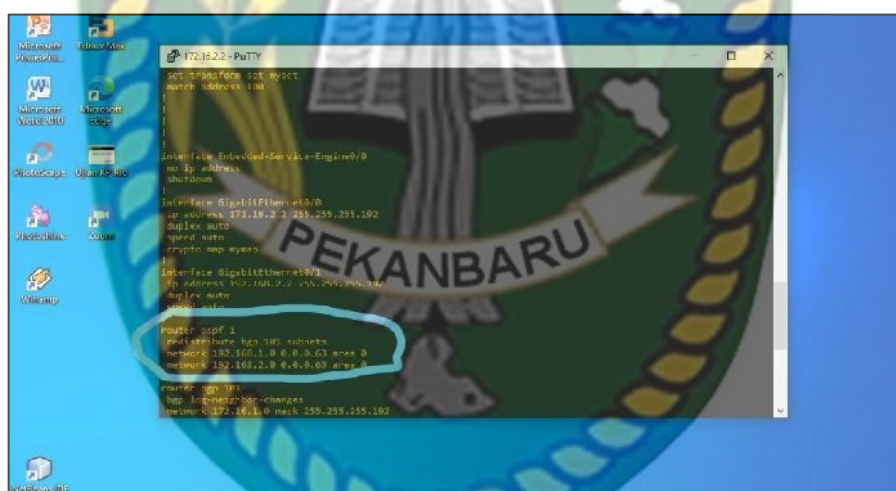
Dari gambar 3.12 diatas, terlihat bahwa penulis mengkonfigurasi protokol OSPF dengan jaringan 192.168.1.0 dan 192.168.2.0 dengan area 0.

Selain melakukan konfigurasi pada Router_OSPF, penulis juga melakukan konfigurasi pada Router_Redistribusi. Konfigurasi yang dilakukan adalah *interface* dan protokol OSPF. Gambar 3.13 dan 3.14 berikut menampilkan secara berurut konfigurasi *interface* dan protokol OSPF pada Router_Redistribusi.



Gambar 3. 13 Ip Interface pada Router_Redistribusi.

Dari gambar 3.13 diatas terlihat penulis mengkonfigurasi ip pada *interface* GE0/1 yaitu 192.168.2.2.



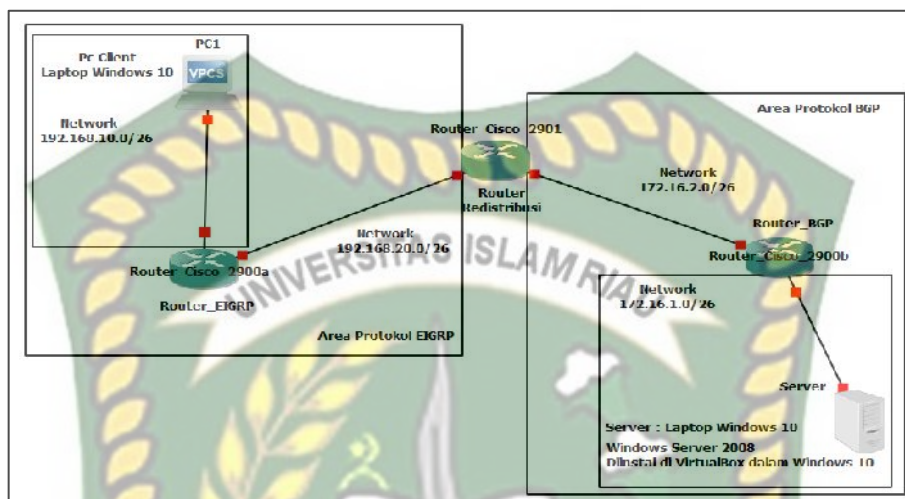
Gambar 3. 14 Konfigurasi Protokol OSPF pada Router_Redistribusi.

Dari gambar 3.14 diatas, terlihat bahwa penulis mengkonfigurasi protokol OSPF dengan jaringan 192.168.1.0 dan 192.168.2.0 dengan area 0.

3.3.2 Jaringan Protokol EIGRP

Pada jaringan EIGRP ini, terdapat 1 buah *router* yang penulis namakan Router_EIGRP. *Router* ini berfungsi penghubung antara jaringan 192.168.10.0/26 dengan 192.168.20.0/26 dimana dari jaringan 192.168.20.0 inilah yang juga akan

menghubungkan jaringan 192.168.10.0 dengan *router* redistribusi. Gambar 3.15 berikut menggambarkan topologi jaringan EIGRP yang penulis rancang.



Gambar 3. 15 Jaringan Topologi Protokol EIGRP.

Dari gambar 3.15 diatas terdapat sebuah pc dimana pada perangkat nyata yang penulis gunakan adalah sebuah laptop dengan sistem operasi Windows 10 yang penulis set sebagai PC klien. Gambar 3.16 berikut merupakan tampilan perangkat nyata yang penulis gunakan sebagai perangkat klien.



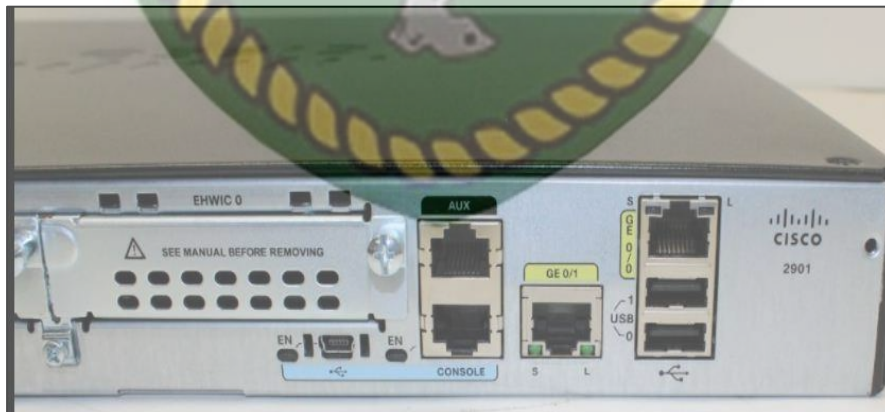
Gambar 3. 16 Perangkat Nyata Laptop Klien pada EIGRP.

Untuk konfigurasi ip pada perangkat klien, penulis melakukan DHCP dengan melakukan konfigurasi pada Router_EIGRP. Untuk pengaturan ip *interface* (kabel) pada Router_EIGRP dapat dilihat pada gambar 3.17 berikut.



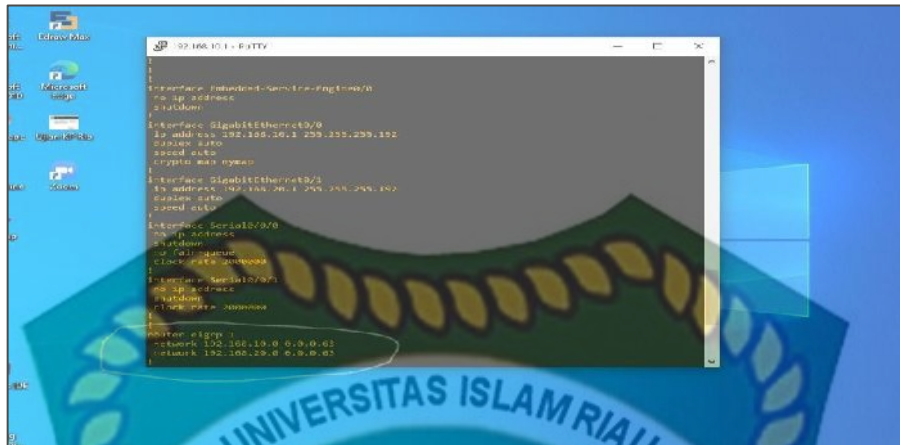
Gambar 3. 17 Konfigurasi IP Interface Router_EIGRP.

Pada gambar 3.17 diatas, *interface* (slot kabel) yang digunakan pada Router_OSPF adalah *interface* GigabitEthernet0/0 (GE0/0) dengan ip 192.168.10.1 dan *interface* GigabitEthernet0/1 (GE0/1) dengan ip 192.168.20.1. Gambar 3.18 berikut ini menampilkan slot pada Router_EIGRP.



Gambar 3. 18 Slot GE0/0 dan GE0/1 pada Router_EIGRP.

Pada Router_EIGRP, penulis melakukan konfigurasi EIGRP. Gambar 3.19 berikut ini menampilkan konfigurasi protokol EIGRP di Router_EIGRP.



```

172.168.10.1 - PuTTY
Interface FastEthernet0/0
 ip address 192.168.10.1 255.255.255.252
 speed auto
 duplex auto

Interface GigabitEthernet0/1
 ip address 192.168.20.1 255.255.255.252
 speed auto

Interface Serial0/0/0
 ip address
 shutdown
 clock rate 2000000
!
Interface Serial0/0/1
 ip address
 shutdown
 clock rate 2000000

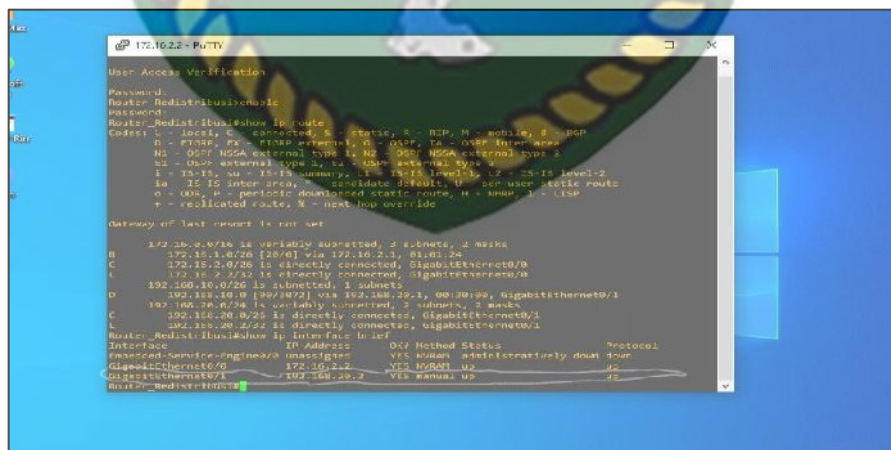
Router eigrp 1
 network 192.168.10.0 0.0.0.01
 network 192.168.20.0 0.0.0.01

```

Gambar 3. 19 Konfigurasi Protokol EIGRP pada Router_EIGRP.

Dari gambar 3.12 diatas, terlihat bahwa penulis mengkonfigurasi protokol EIGRP dengan jaringan 192.168.10.0 dan 192.168.20.0 .

Selain melakukan konfigurasi pada Router_EIGRP, penulis juga melakukan konfigurasi routing protokol EIGRP pada Router_Redistribusi. Konfigurasi yang dilakukan adalah *interface* dan protokol EIGRP. Gambar 3.20 dan 3.21 berikut menampilkan secara berurut konfigurasi *interface* dan protokol EIGRP pada Router_Redistribusi.



```

172.16.22 - PuTTY
User Access Verification
Password:
Router Redistribusi>enable
Router Redistribusi>show ip route
Codes: C - local, L - learned, S - static, R - RIP, M - mobile, B - BGP
       O - OSPF, EX - OSPF external, E - EIGRP, E2 - EIGRP external
       N1 - NLSR, N2 - NLSR summary, IN - INRIA level 1, I2 - INRIA level 2
       IA - IS-IS inter area, * candidate default, U - user defined route
       O - ODR, P - periodic downloaded static route, H - HSRP, I - ISIS
       * - replicated route, % - next hop override

Gateway of last resort is not set

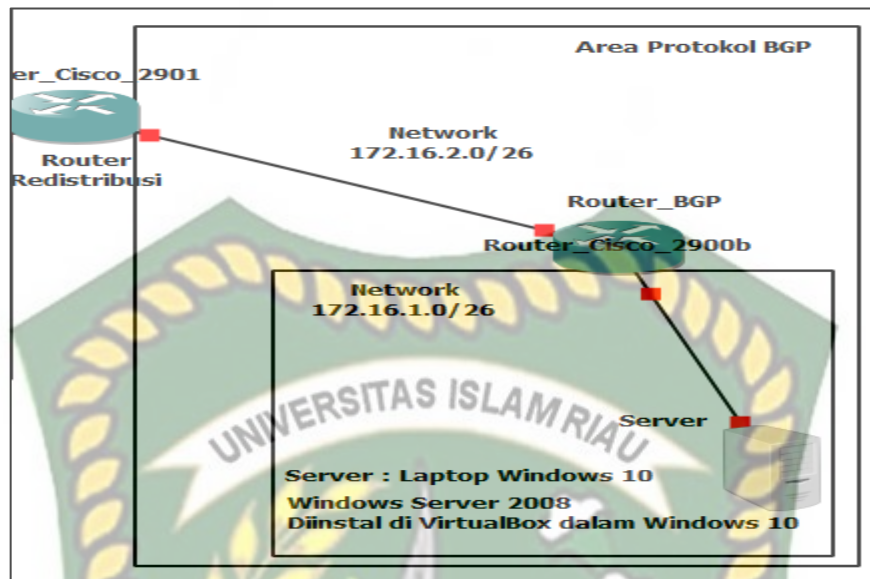
 172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
  B   172.16.1.0/20 [10/0] via 172.16.21.1, GigabitEthernet0/0
  C   172.16.2.0/26 is directly connected, GigabitEthernet0/0
  C   192.168.10.0/26 is subnetted, 1 subnets
  D   192.168.10.0 [90/1072] via 192.168.20.1, GigabitEthernet0/1
  C   192.168.20.0/24 is variably subnetted, 2 subnets, 2 masks
  C   192.168.20.0/26 is directly connected, GigabitEthernet0/1
  L   192.168.20.2/32 is directly connected, GigabitEthernet0/1

Router Redistribusi>show ip interface brief
Interface              IP Address         O/M  Method Status          Protocol
FastEthernet0/0        unassigned        YES  manual administratively down  down
GigabitEthernet0/0     172.16.2.2        YES  manual UP           IP
GigabitEthernet0/1     192.168.20.2      YES  manual UP           IP
Router Redistribusi>

```

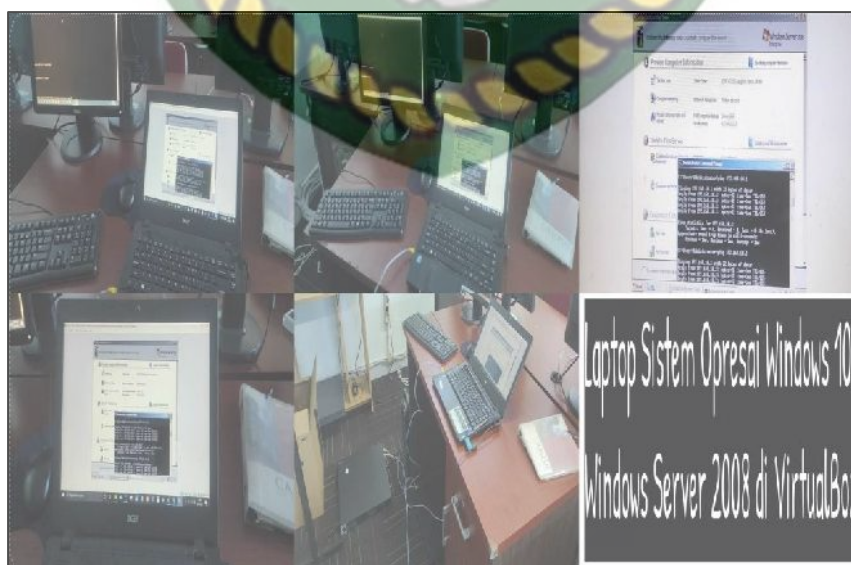
Gambar 3. 20 Ip Interface Pada GE0/1 Router_Redistribusi.

Dari gambar 3.20 diatas terlihat penulis mengkonfigurasi ip pada *interface* GE0/1 yaitu 192.168.20.2.



Gambar 3. 22 Topologi Jaringan Protokol BGP.

Dari gambar 3.22 diatas terdapat sebuah *server* dimana pada perangkat nyata yang penulis gunakan adalah sebuah laptop dengan sistem operasi Windows 10 dan menginstal windows server 2008 didalamnya dalam bentuk OS virtual dengan menggunakan VirtualBox yang penulis set sebagai *server*. Gambar 3.23 berikut merupakan tampilan perangkat nyata yang penulis gunakan sebagai perangkat server.



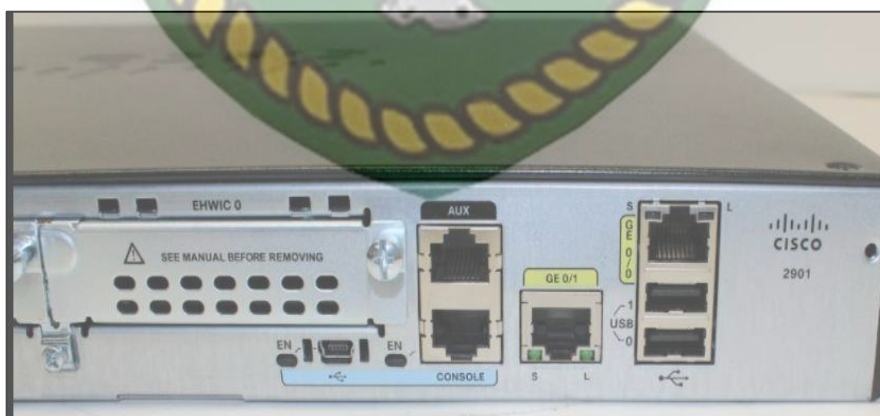
Gambar 3. 23 Laptop Acer sebagai Server di Jaringan BGP.

Untuk konfigurasi ip pada perangkat *server*, penulis melakukan DHCP dengan melakukan konfigurasi pada Router_BGP. Untuk pengaturan ip *interface* (kabel) pada Router_BGP dapat dilihat pada gambar 3.24 berikut.



Gambar 3. 24 Konfigurasi IP Interface pada Router_BGP.

Pada gambar 3.10 diatas, *interface* (slot kabel) yang digunakan pada Router OSPF adalah *interface* GigabitEthernet0/0 (GE0/0) dengan ip 172.16.1.1 dan *interface* GigabitEthernet0/1 (GE0/1) dengan ip 172.16.2.1. Gambar 3.25 berikut ini menampilkan slot pada Router_BGP.



Gambar 3. 25 Slot GE0/0 dan GE0/1 pada Router_BGP.

Pada Router_BGP, penulis melakukan konfigurasi BGP dengan nilai AS 100. Gambar 3.26 berikut ini menampilkan konfigurasi protokol BGP di Router_BGP.

Dari gambar 3.27 diatas terlihat penulis mengkonfigurasi ip pada *interface* GE0/0 yaitu 172.16.2.2.



Gambar 3. 28 Konfigurasi Protokol BGP pada Router_Redistribusi.

Dari gambar 3.28 diatas, terlihat bahwa penulis mengkonfigurasi protokol BGP dengan jaringan 172.16.1.0 dan 172.16.2.0 dengan nilai AS 101 dan menetapkan ip 172.16.2.1 sebagai *neighbor* dengan *remote-as* 100 dimana ip milik ini berada pada *interface* GE0/1 di Router_BGP.

3.4 Proses Redistribusi (*Redistribution*) Protokol

Pada penelitian ini penulis membangun sebuah jaringan komputer yang menghubungkan 2 buah jaringan dengan protokol *routing* dinamis yang berbeda. Untuk menghubungkan dua jaringan yang memiliki protokol berbeda, digunakan sebuah metode yang disebut dengan redistribusi (*redistribution*). Pada penelitian ini digunakan 3 protokol *routing* dinamis, yaitu OSPF, EIGRP dan BGP yang menjadi protokol yang penulis analisa.

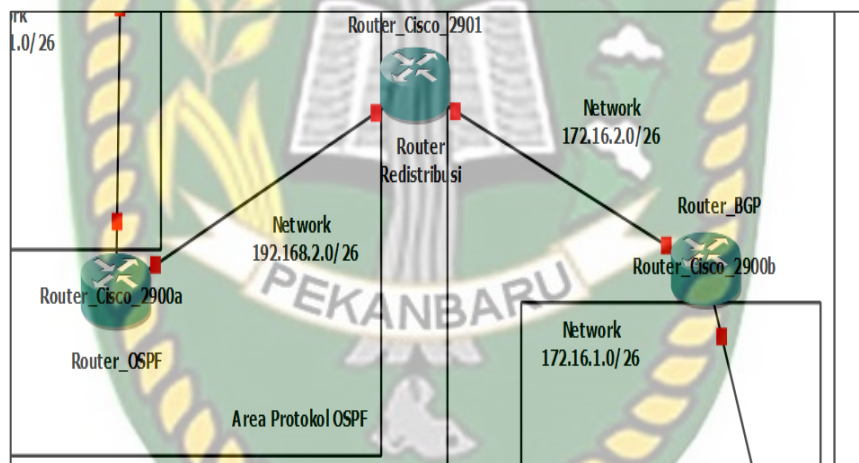
Dalam penelitian ini, penulis membagi proses redistribusi menjadi 2 ,yaitu

1. Redistribusi jaringan antara OSPF dan BGP.

2. Redistribusi jaringan antara EIGRP dan BGP.

3.4.1 Redistribusi Jaringan antara OSPF dan BGP

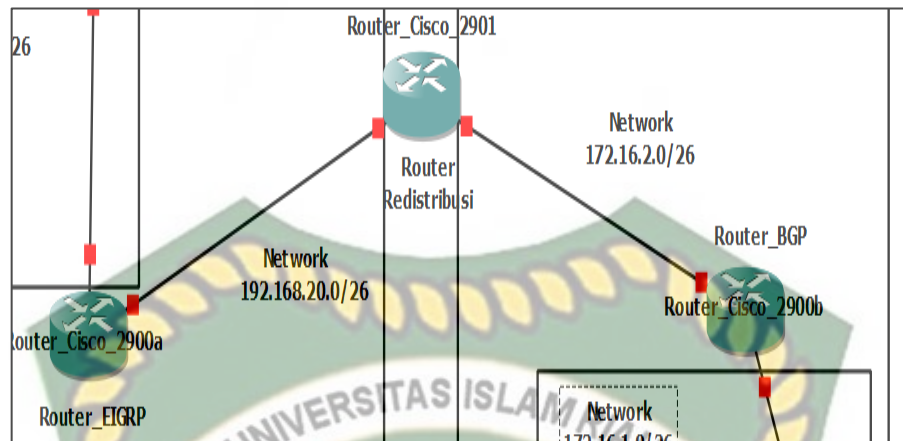
Pada proses redistribusi jaringan antara BGP dan OSPF ini, peneliti merancang skema dengan membuat 2 buah *network* penghubung jaringan OSPF dan BGP menuju *router* redistribusi. *Network* penghubung jaringan OSPF menuju *router* redistribusi adalah 192.168.2.0 dan *network* penghubung jaringan BGP menuju *router* redistribusi adalah 172.16.2.0. Gambar 3.29 berikut menampilkan *network* redistribusi antara OSPF dan BGP.



Gambar 3. 29 Redistribusi antara OSPF dan BGP.

3.4.2 Redistribusi Jaringan antara EIGRP dan BGP

Pada proses redistribusi jaringan antara EIGRP dan BGP ini, peneliti merancang skema dengan membuat 2 buah *network* penghubung jaringan EIGRP dan BGP menuju *router* redistribusi. *Network* penghubung jaringan EIGRP menuju *router* redistribusi adalah 192.168.20.0 dan *network* penghubung jaringan BGP menuju *router* redistribusi adalah 172.16.2.0. Gambar 3.30 berikut menampilkan *network* redistribusi antara EIGRP dan BGP.



Gambar 3. 30 Redistribusi antara EIGRP dan BGP.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Pengujian Sistem

Proses analisa penelitian kinerja redistribusi *routing* dinamis antara protokol OSPF, EIGRP dan BGP ini, dirancang sebuah jaringan dengan membangun 3 buah topologi dengan protokol masing – masing yaitu OSPF, EIGRP dan BGP. Ketiga topologi tersebut akan dihubungkan dengan sebuah *router* sebagai penghubung agar bisa berkomunikasi dengan menerapkan metode redistribusi.

Pada penelitian ini akan melakukan analisa terhadap kinerja redistribusi antara protokol OSPF dengan BGP dan EIGRP dan BGP berdasarkan nilai *throughput*, *packet loss* dan *delay*. Setiap perangkat akhir akan diuji koneksinya masing – masing. Setelah setiap perangkat akhir sudah bisa terkoneksi dengan baik maka akan dilakukan proses pencarian nilai *throughput*, *packet loss* dan *delay*.

Paket yang dikirim adalah ICMP dengan cara *ping* dari perangkat akhir di topologi protokol BGP ke perangkat akhir OSPF untuk proses redistribusi BGP ke OSPF dan perangkat akhir di topologi protokol BGP ke EIGRP untuk proses redistribusi BGP ke EIGRP.

4.2 Proses *Capture* paket ICMP menggunakan Wireshark

Pada penelitian ini, proses pencarian nilai *throughput*, *packet loss* dan *delay* dilakukan dengan melakukan *ping* dari komputer yang ada pada jaringan protokol

BGP menuju server yang ada pada protokol OSPF dan protokol EIGRP. Gambar 4.1 berikut ini menampilkan proses ping yang dilakukan.

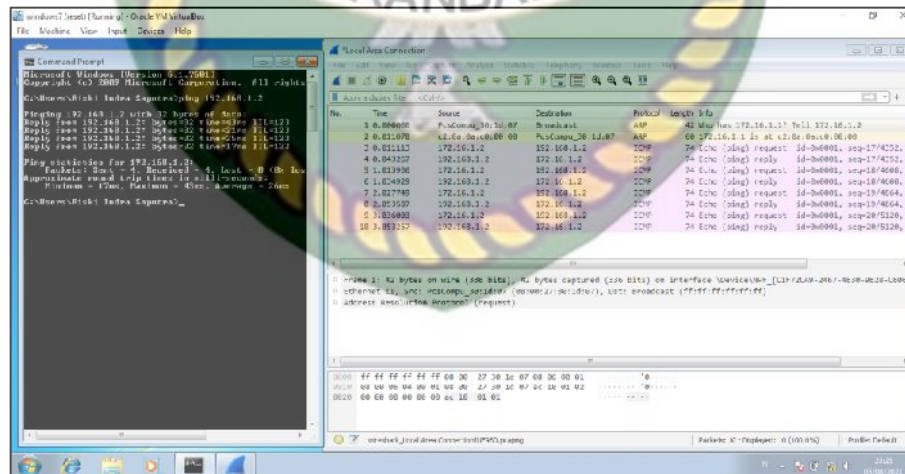
```
C:\Users\Riski Indra Saputra>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:
Reply from 192.168.1.2: bytes=32 time=22ms TTL=123
Reply from 192.168.1.2: bytes=32 time=17ms TTL=123
Reply from 192.168.1.2: bytes=32 time=16ms TTL=123
Reply from 192.168.1.2: bytes=32 time=14ms TTL=123

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 14ms, Maximum = 22ms, Average = 17ms
```

Gambar 4. 1 Proses PING.

Berdasarkan gambar 4.1 diatas, banyak paket yang dikirim 1 kali proses ping adalah 4 dengan besar paket sebesar 32 bit. Pada saat bersamaan dengan proses ping dilakukan, aplikasi wireshark diaktifkan untuk melakukan capture data paket ICMP. Gambar 4.2 berikut ini menampilkan proses capture paket ICMP pada wireshark.



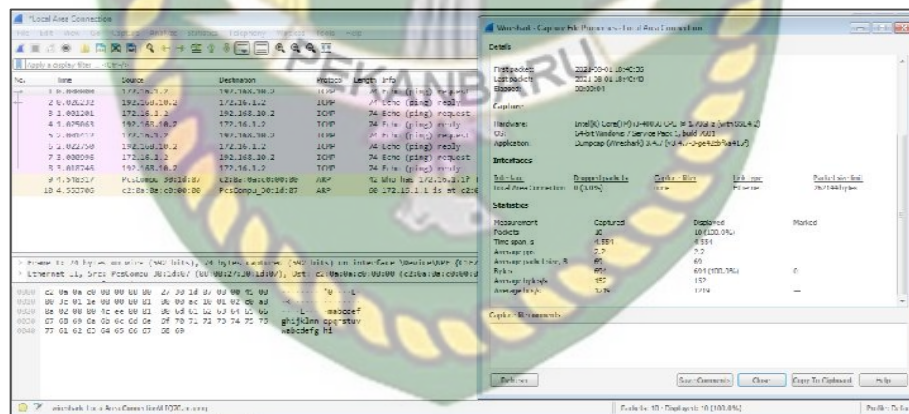
Gambar 4. 2 Proses Capture Paket ICMP pada Wireshark.

Berdasarkan gambar 4.2 diatas, 1 kali proses ping yang dilakukan dengan jumlah paket ICMP yang dikirim sebanyak 4 maka pada wireshark ditampilkan jumlah ICMP sebanyak 8. Hal ini dikarenakan adanya proses request dan replay.

Proses *request* merupakan permintaan berkomunikasi oleh perangkat *source* berupa paket yang dikirim kepada perangkat *destination* sedangkan *reply* merupakan paket yang dikirim oleh perangkat *destination* sebagai respon atas permintaan berkomunikasi dari perangkat *source*. Dari gambar diatas terlihat jumlah *request* ada 4 dan *reply* juga ada 4 sehingga total dari inilah menampilkan jumlah paket ICMP ada 8.

4.2.1 Proses Mendapatkan Nilai *Throughput*

Setelah terasa cukup penangkapan paket ICMP menggunakan wireshark, maka lakukan *stop capture* yang terdapat pada menu wireshark. Lalu klik menu *statistic* dan pilih *capture file properties*. Gambar 4.3 berikut ini menampilkan *form capture file properties*.



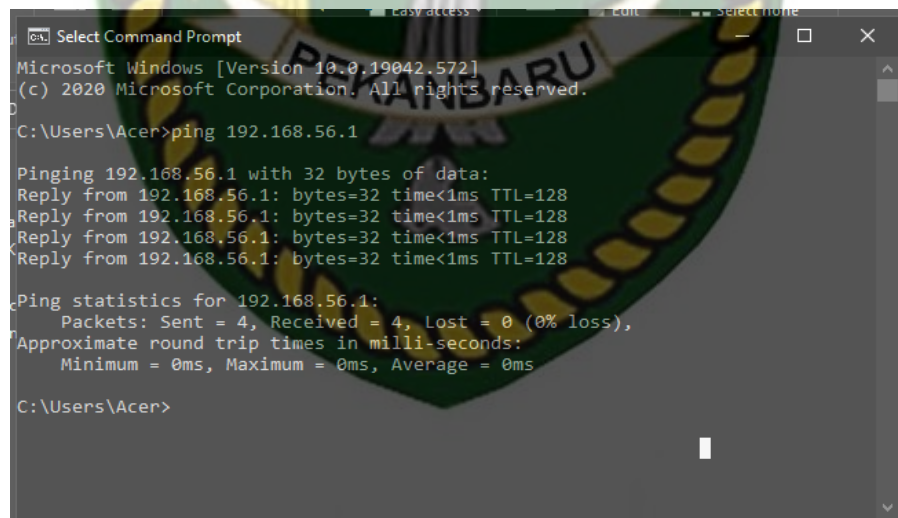
Gambar 4. 3 Proses Mendapatkan Nilai *Throughput*.

Berdasarkan gambar 4.3 diatas, didapatkan nilai *throughput* sebesar 1219 bit/s. Secara teori untuk mendapatkan nilai *throughput* dapat dengan menggunakan rumus. Dari gambar 4.3 diatas, untuk jumlah data yang dikirim ditandai dengan *bytes* dengan nilai sebesar 694. Waktu pengiriman ditandai dengan *Time span, s* dengan nilai 4.554 s. Maka, berdasarkan rumus diperoleh nilai *throughput* = 694

bytes : $4.554 \text{ s} = 152.36 \text{ bytes/s}$. Nilai 152 masih dalam bentuk satuan *bytes* sehingga perlu dikonversi menjadi bit. Berdasarkan teori, $1 \text{ bytes} = 8 \text{ bit}$. Dengan demikian nilai 152.35 bytes/s dikonversi menjadi bit/s harus dikalikan 8 bit maka didapatkan nilai sebesar $152.36 \times 8 = 1218.88 \text{ bit/s}$ atau dibulatkan menjadi 1219 bit/s. Dengan demikian nilai *throughput* yang dihasilkan aplikasi wireshark sama dengan rumus persamaan *throughput* secara teori.

4.2.2 Proses Mendapatkan Nilai *Packet Loss*

Packet loss merupakan jumlah paket yang gagal dikirim oleh perangkat sumber ketika membangun komunikasi sehingga tidak ada respon dari perangkat tujuan. *Packet loss* dapat dilihat dari proses *ping* cmd, gambar 4.4 dan gambar 4.5 berikut ini menampilkan proses *ping* dan hasil *paket loss* yang terjadi.



```

Select Command Prompt
Microsoft Windows [Version 10.0.19042.572]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Acer>ping 192.168.56.1

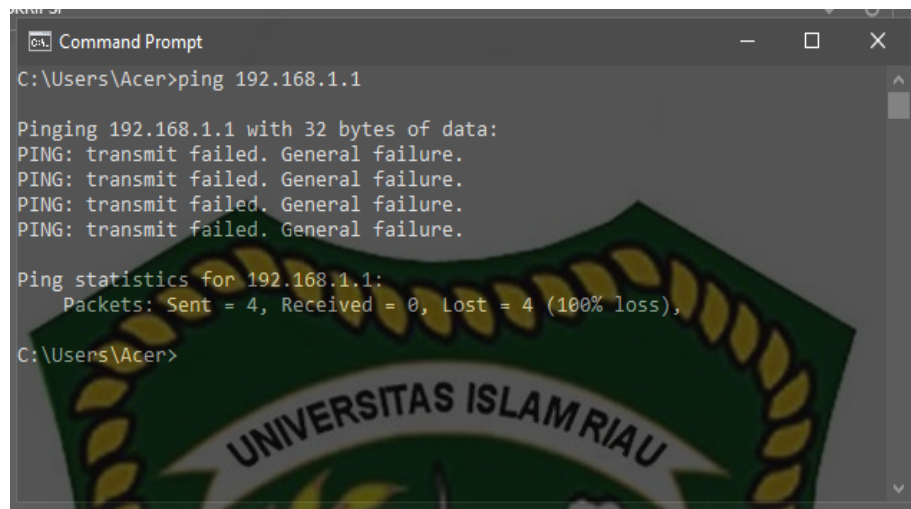
Pinging 192.168.56.1 with 32 bytes of data:
Reply from 192.168.56.1: bytes=32 time<1ms TTL=128
Reply from 192.168.56.1: bytes=32 time<1ms TTL=128
Reply from 192.168.56.1: bytes=32 time<1ms TTL=128
Reply from 192.168.56.1: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.56.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Acer>
  
```

Gambar 4. 4 Ping Packet Loss 0%.

Pada gambar 4.4 terlihat nilai *paket loss* dari 4 paket yang dikirim 0%. Hal ini terjadi karena seluruh paket yang dikirim berhasil diterima.



```

Command Prompt
C:\Users\Acer>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:
PING: transmit failed. General failure.
PING: transmit failed. General failure.
PING: transmit failed. General failure.
PING: transmit failed. General failure.

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\Users\Acer>

```

Gambar 4.5 Ping Packet Loss 100%.

Sementara pada gambar 4.5 nilai *paket loss* yang didapat 100%, karena ke 4 paket yang dikirim gagal diterima. Secara teori *packet loss* dapat dicari dengan menggunakan rumus. Berdasarkan gambar 4.4 jika mencari nilai *packet loss* menggunakan rumus, maka $packet\ loss = \left(\frac{4-4}{4}\right) \times 100\% = 0\%$. Sedangkan untuk gambar 4.5 dimasukkan dalam rumus secara teori, maka $packet\ loss = \left(\frac{4-0}{4}\right) \times 100\% = 100\%$.

4.2.3 Proses Mendapatkan Nilai *Delay*

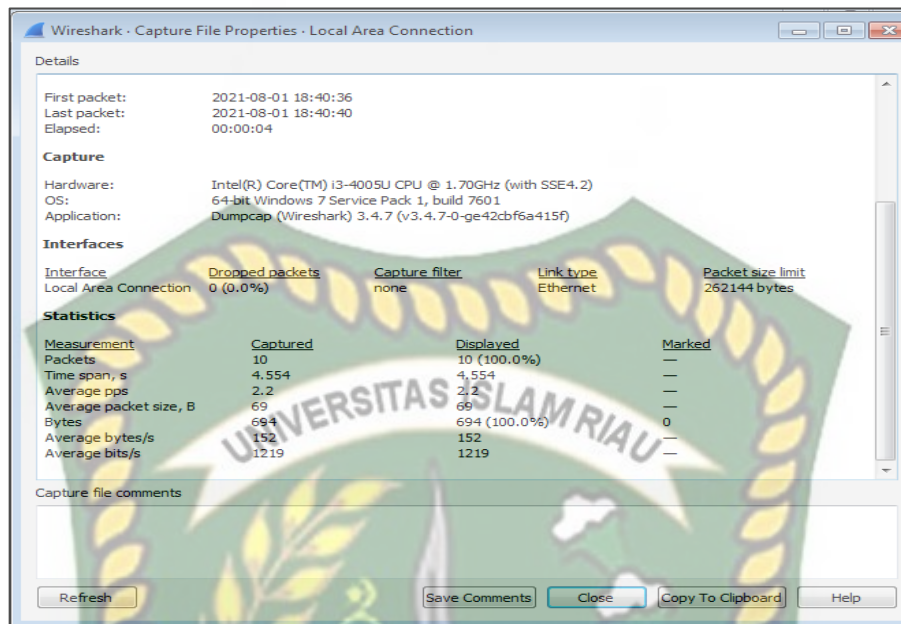
Proses untuk mendapatkan nilai *delay* dilakukan dengan cara menyimpan file hasil *captured* *wirshark* dengan ekstensi *.csv*. File ini dibuka dengan aplikasi *microsoft excel*. Gambar 4.6 dibawah ini menampilkan file *csv* yang dibuka dengan *microsoft excel*.

| No. | Time | Source | Destination | Protocol | Length | Info | Time 2 | Time 1 | Delay |
|-----|----------|--|--|----------|--------|--|-------------|----------|----------|
| 1 | 0,006292 | 172.16.1.2 | 152.168.10.2 | ICMP | 74 | Echo (ping) request Id=0x0001, seq=10/27934, ttl=128 (request in 2) | 0,006292 | 0 | 0,006292 |
| 2 | 0,006292 | 152.168.10.2 | 172.16.1.2 | ICMP | 74 | Echo (ping) reply Id=0x0001, seq=10/27934, ttl=128 (reply in 1) | 1,001201 | 0,006292 | 0,994909 |
| 3 | 1,001201 | 172.16.1.2 | 152.168.10.2 | ICMP | 74 | Echo (ping) request Id=0x0001, seq=110/28160, ttl=128 (request in 4) | 1,005063 | 1,001201 | 0,003862 |
| 4 | 1,005063 | 152.168.10.2 | 172.16.1.2 | ICMP | 74 | Echo (ping) reply Id=0x0001, seq=110/28160, ttl=128 (reply in 3) | 2,001412 | 1,005063 | 0,996349 |
| 5 | 2,001412 | 172.16.1.2 | 152.168.10.2 | ICMP | 74 | Echo (ping) request Id=0x0001, seq=111/28416, ttl=128 (request in 6) | 2,002275 | 2,001412 | 0,000863 |
| 6 | 2,002275 | 152.168.10.2 | 172.16.1.2 | ICMP | 74 | Echo (ping) reply Id=0x0001, seq=111/28416, ttl=128 (reply in 5) | 3,000996 | 2,002275 | 0,998721 |
| 7 | 3,000996 | 172.16.1.2 | 152.168.10.2 | ICMP | 74 | Echo (ping) request Id=0x0001, seq=112/28672, ttl=128 (request in 8) | 3,018796 | 3,000996 | 0,017800 |
| 8 | 3,018796 | 152.168.10.2 | 172.16.1.2 | ICMP | 74 | Echo (ping) reply Id=0x0001, seq=112/28672, ttl=128 (reply in 7) | Total Delay | | 3,018796 |
| 9 | 4,48317 | Hostcompu_302ad0c220a0a0000000000000000000 | Hostcompu_302ad0c220a0a0000000000000000000 | HTTP | 42 | Who has 172.16.1.1? Tell 172.16.1.2 | Host2 Delay | | 0,00581 |
| 10 | 4,48317 | Hostcompu_302ad0c220a0a0000000000000000000 | Hostcompu_302ad0c220a0a0000000000000000000 | HTTP | 70 | 172.16.1.1 is at 220a0a00000000000000000000000000 | | | |

Gambar 4. 6 Proses Mendapatkan Nilai Delay.

Berdasarkan gambar 4.6 diatas, untuk mendapatkan nilai *delay* maka dilakukan perhitungan. *Delay* merupakan selisih waktu perangkat sumber melakukan *request* ke perangkat akhir dengan respon (*replay*) yang perangkat akhir lakukan terhadap *request* dari perangkat sumber. Dari gambar 4.6 diatas paket ICMP ada 8. Pada penelitian ini selisih waktu yang dihitung adalah waktu yang diperlukan oleh paket ICMP untuk melakukan *request* dan *replay* dengan persamaan $delay = time\ 2 - time\ 1$.

Total *delay* merupakan jumlah dari hasil seluruh *delay*. Sedangkan rata-rata *delay* merupakan hasil dari total *delay* dibagi dengan jumlah *bytes* paket yang dikirim. Jumlah *byts* paket yang dikirim dapat dilihat pada *form capture file properties* pada wireshark dengan cara lakukan *stop capture* yang terdapat pada menu wireshark. Lalu klik menu *statistic* dan pilih *capture file properties*. Gambar 4.7 menampilkan *form file properties*.



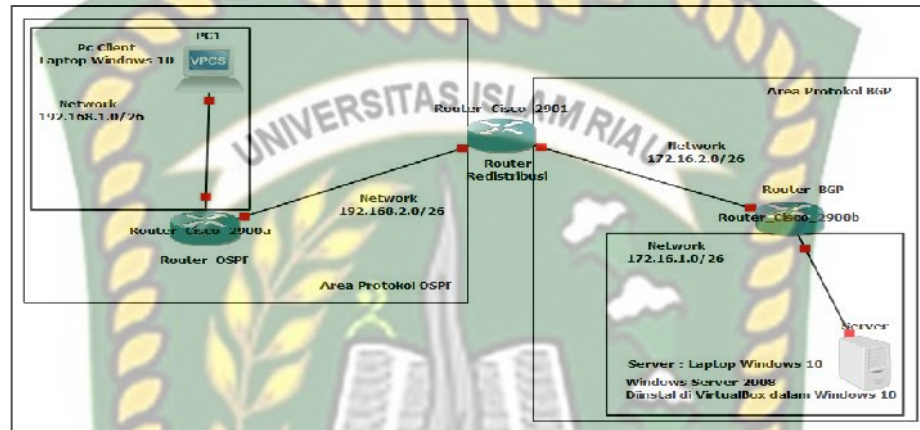
Gambar 4.7 Form File Properties.

Dari gambar 4.7 diatas, terlihat jumlah *bytes* paket yang dikirim sebesar 694. Dengan demikian, untuk mencari nilai dari rata – rata *delay* dapat dihitung dengan rata – rata $delay = total\ delay / 694$.

4.3 Skema Analisa *Throughput*, *Packet Loss* dan *Delay* Redistribusi OSPF dan BGP

Pada topologi jaringan BGP terdapat sebuah *server* dengan sistem operasi Windows Server 2008 yang diinstall dalam bentuk OS virtual dalam windows 10 dengan bantuan VirtualBox yang berfungsi sebagai perangkat tujuan dan di topologi OSPF terdapat sebuah klien berupa laptop dengan sistem operasi Windows 10 sebagai perangkat sumber ketika melakukan *ping* untuk menganalisa nilai *throughput*, *packet loss* dan *delay* paket ICMP pada redistribusi OSPF dan BGP.

Aplikasi wireshark dipasang pada komputer yang ada pada jaringan OSPF. Dengan menggunakan aplikasi ini peneliti mencari nilai *throughput*, *packet loss* dan *delay*. Gambar 4.8 berikut menampilkan topologi protokol BGP dan OSPF yang didistribusi.



Gambar 4. 8 Topologi Jaringan Redistribusi OSPF dan BGP.

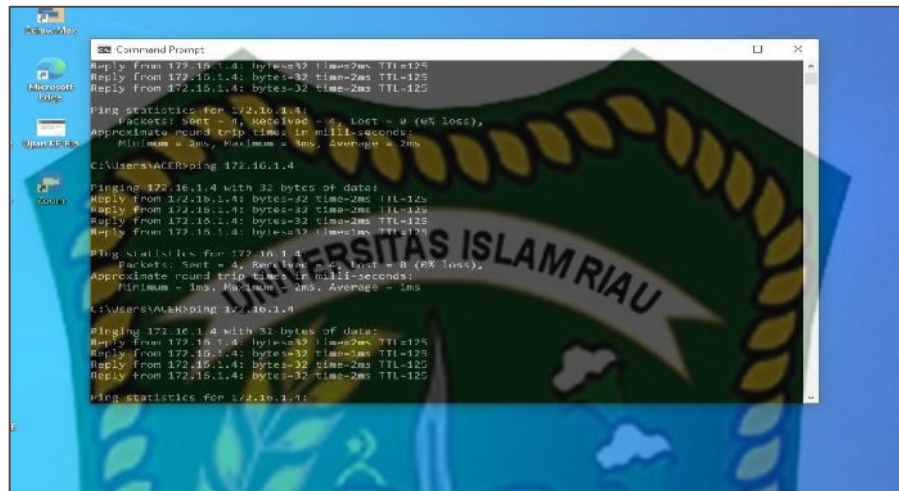
Berdasarkan gambar 4.8 diatas, area OSPF memiliki 2 *network* diantaranya *network* 192.168.1.0, *network* 192.168.2.0. Untuk area BGP juga memiliki 2 *network* diantaranya 172.16.1.0 dan 172.16.2.0.

4.3.1 Proses Analisa Nilai *Throughput* Redistribusi OSPF dan BGP

Proses pencarian nilai *throughput* yang diterapkan adalah dengan melakukan penangkapan paket ICMP melalui *ping* pada (*command prompt*) cmd dengan bantuan aplikasi wireshark. Aplikasi wireshark dipasang pada komputer yang ada pada area OSPF dan komputer ini bertindak sebagai perangkat sumber dalam melakukan *ping*.

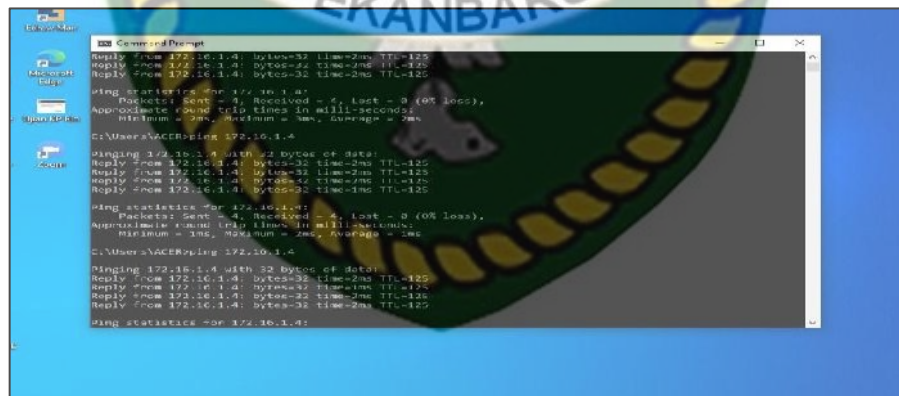
Berdasarkan gambar 4.8, komputer ini diberi nama dengan PC1. Perangkat yang ada pada area OSPF yang diberi nama Server digunakan sebagai perangkat tujuan. Pada penelitian ini perangkat sumber (PC1) diberi ip 192.168.1.2 dan

perangkat tujuan (Server) diberi ip 172.168.1.4. Gambar 4.9 berikut menampilkan terkoneksiya perangkat sumber dan perangkat tujuan.



Gambar 4. 9 Koneksi Perangkat Sumber menuju Perangkat Tujuan.

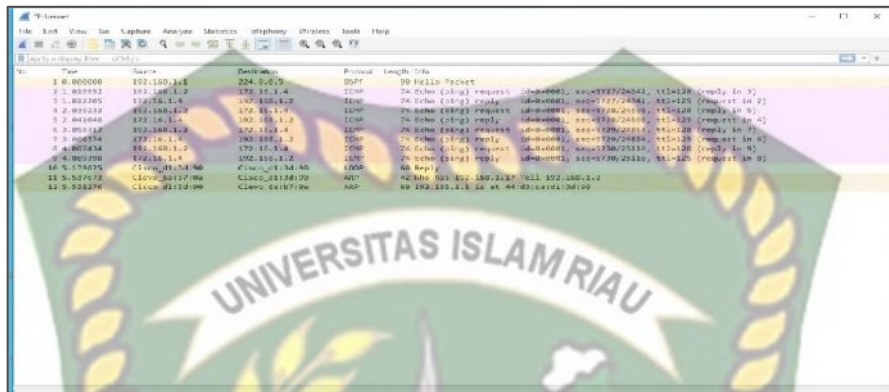
Setelah perangkat sumber dan tujuan terkoneksi, langkah selanjutnya melakukan *ping* untuk mencari nilai *throughput*. Gambar 4.10 menampilkan proses *ping* dari perangkat menuju perangkat tujuan (Server di BGP).



Gambar 4. 10 Ping dari Laptop di OSPF (Windows 10) menuju Server di BGP.

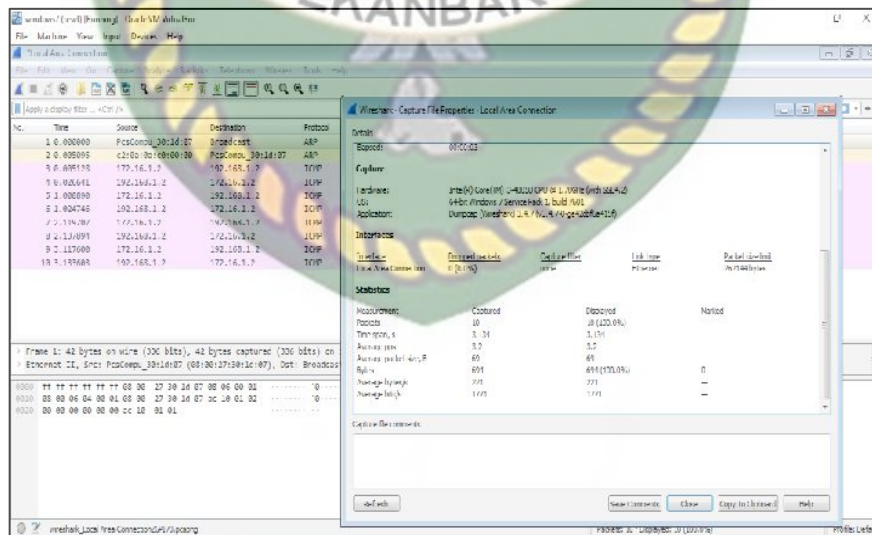
Pada saat bersamaan *ping* dilakukan, aplikasi wireshark aktif dan melakukan *capture* data paket ICMP. Batasan proses penangkapan data paket ICMP oleh wireshark adalah 1 kali ping dengan jumlah paket yang dikirim

sebanyak 4 . Gambar 4.11 menampilkan proses *ping* sekaligus penangkapan paket data ICMP oleh wireshark.



Gambar 4. 11 Penangkapan Paket ICMP oleh Wireshark pada Redistribusi OSPF dan BGP.

Setelah melakukan pemberhentian *capture* data paket ICMP karena paket ICMP yang tercapture sudah memenuhi jumlah *request* dan *replay* 1 kali *ping*. Selanjutnya adalah menganalisa *capture file properties*. Gambar 4.12 berikut menampilkan hasil *capture file properties* pada redistribusi OSPF dan BGP.



Gambar 4. 12 Tampilan *Capture File Properties* 1 kali *Ping* pada Redistribusi BGP dan OSPF.

Berdasarkan gambar 4.12 diatas, didapatkan nilai *throughput* 1 kali *ping* dengan jumlah paket yang dikirim sebanyak 4 sebesar 1771 bit/s.

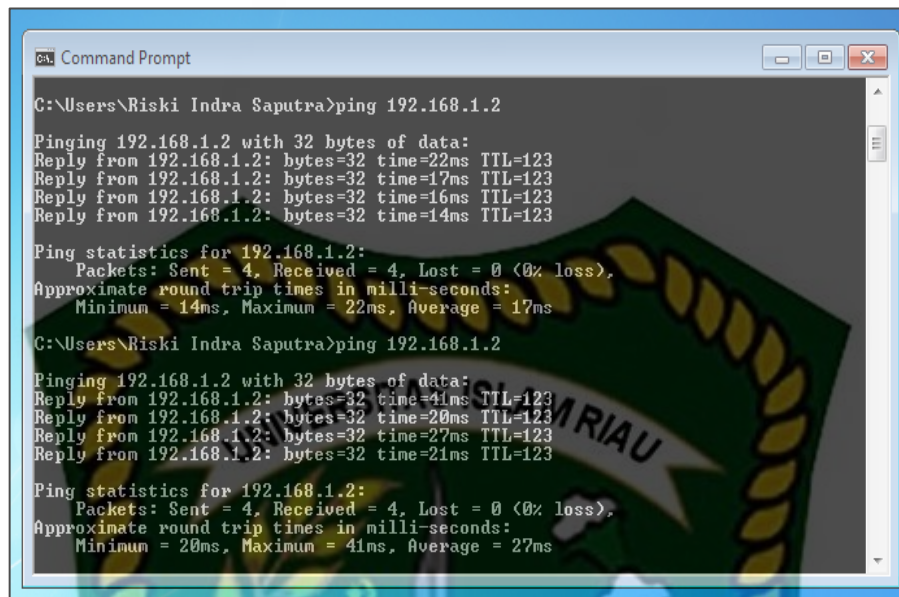
Pada penelitian ini, proses *ping* dilakukan sebanyak 10 kali dimana 1 kali *ping* jumlah paket yang dikirim sebanyak 4 dan didapatkan nilai *throughput* seperti keterangan yang ditampilkan tabel 4.1 berikut :

Tabel 4. 1 Nilai *Throughput* Redistribusi BGP dan OSPF.

| NO. | Proses <i>Ping</i> | Nilai <i>Throughput</i> (bytes/s) |
|-----|--------------------|--------------------------------------|
| 1 | <i>Ping</i> ke -1 | 1771 |
| 2 | <i>Ping</i> ke -2 | 1818 |
| 3 | <i>Ping</i> ke -3 | 1777 |
| 4 | <i>Ping</i> ke -4 | 1825 |
| 5 | <i>Ping</i> ke -5 | 1835 |
| 6 | <i>Ping</i> ke -6 | 1219 |
| 7 | <i>Ping</i> ke -7 | 1077 |
| 8 | <i>Ping</i> ke -8 | 1094 |
| 9 | <i>Ping</i> ke -9 | 1343 |
| 10 | <i>Ping</i> ke -10 | 1438 |

4.3.2 Proses Analisa Nilai *Packet Loss* Redistribusi OSPF dan BGP

Proses mendapatkan nilai *packet loss* pada penelitian ini berdasarkan proses *ping* diambil dengan cara hasil tampilan *command prompt* (cmd) dan *capture file properties* milik wireshark. Gambar 4.13 menampilkan *packet loss* yang dihasilkan *ping* pada redistribusi OSPF dan BGP pada *command prompt* (cmd).



```

C:\Users\Riski Indra Saputra>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:
Reply from 192.168.1.2: bytes=32 time=22ms TTL=123
Reply from 192.168.1.2: bytes=32 time=17ms TTL=123
Reply from 192.168.1.2: bytes=32 time=16ms TTL=123
Reply from 192.168.1.2: bytes=32 time=14ms TTL=123

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 14ms, Maximum = 22ms, Average = 17ms

C:\Users\Riski Indra Saputra>ping 192.168.1.2

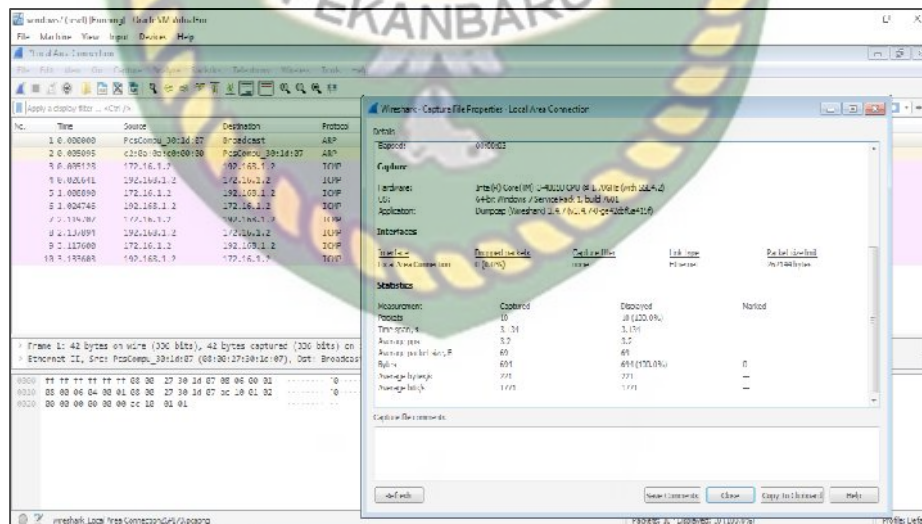
Pinging 192.168.1.2 with 32 bytes of data:
Reply from 192.168.1.2: bytes=32 time=41ms TTL=123
Reply from 192.168.1.2: bytes=32 time=20ms TTL=123
Reply from 192.168.1.2: bytes=32 time=27ms TTL=123
Reply from 192.168.1.2: bytes=32 time=21ms TTL=123

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 20ms, Maximum = 41ms, Average = 27ms

```

Gambar 4.13 Tampilan *Packet Loss* yang dihasilkan *Ping* Pada Redistribusi OSPF dan BGP pada cmd.

Berdasarkan gambar 4.13 diatas, *packet loss* yang didapat adalah 0 (0%). Selanjutnya *packet loss* yang didapatkan dari hasil pada *capture file properties* pada wireshark dapat dilihat pada gambar 4.14 berikut ini.



Gambar 4.14 Nilai *Packet Loss* dari Hasil *Capture File Properties* Pada Wireshark Redistribusi BGP dan OSPF.

Berdasarkan gambar 4.14 diatas, nilai *packet loss* dapat diambil dari nilai *dropped packet*. Pada gambar terlihat nilai *dropped packet* adalah 0 (0%). Dari 2

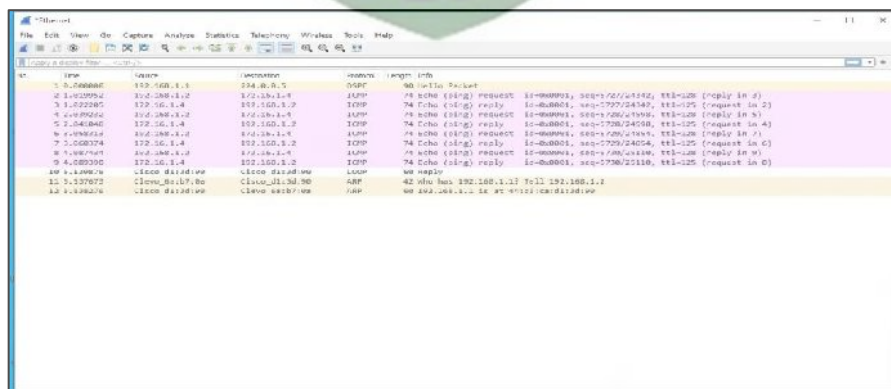
cara ini dapat dilihat bahwa *packet loss* yang dihasilkan cmd dan *capture file properties* pada wireshark adalah sama. Berikut tabel hasil analisa *packet loss* yang dihasilkan 10 kali proses *ping* yang dilakukan pada penelitian ini.

Tabel 4. 2 Nilai *Packet Loss* Redistribusi OSPF dan BGP.

| NO. | Proses <i>Ping</i> | Nilai <i>Packet Loss</i> (%) |
|-----|--------------------|---------------------------------|
| 1 | <i>Ping</i> ke -1 | 0 |
| 2 | <i>Ping</i> ke -2 | 0 |
| 3 | <i>Ping</i> ke -3 | 0 |
| 4 | <i>Ping</i> ke -4 | 0 |
| 5 | <i>Ping</i> ke -5 | 0 |
| 6 | <i>Ping</i> ke -6 | 0 |
| 7 | <i>Ping</i> ke -7 | 0 |
| 8 | <i>Ping</i> ke -8 | 0 |
| 9 | <i>Ping</i> ke -9 | 0 |
| 10 | <i>Ping</i> ke -10 | 0 |

4.3.3 Proses Analisa Nilai *Delay* Redistribusi OSPF dan BGP

Proses pencarian nilai *delay* pada penelitian ini, dilakukan dengan melakukan penyimpanan file hasil *capture* pada wireshark ekstensi csv. Gambar 4.15 berikut menampilkan hasil *capture* paket ICMP dari proses *ping* ke-1 menggunakan wireshark pada redistribusi OSPF dan BGP.



Gambar 4. 15 Tampilan Hasil *Capture Ping* ke-1 Redistribusi OSPF dan BGP.

Hasil dari *capture* tersebut disimpan menjadi file dengan ekstensi csv untuk melakukan perhitungan *delay* yang terjadi pada proses *ping* ke-1 redistribusi OSPF dan BGP. Tampilan file csv dari hasil *capture* paket ICMP *ping* ke-1 redistribusi OSPF dan BGP dapat dilihat pada gambar 4.16 berikut.

| No | Time | Source | Destination | Protocol | Length | Info | Time 1 | Time 2 | Delay |
|-------------|-----------|-------------|-------------|----------|--------|----------------|----------|----------|----------|
| 1 | 0 | 192.168.1.1 | 192.168.1.2 | ICMP | 90 | Echo (request) | 0.000000 | 0.000000 | 0.000000 |
| 2 | 1.0119429 | 192.168.1.2 | 192.168.1.1 | ICMP | 78 | Echo (reply) | 0.000000 | 0.000000 | 0.000000 |
| 3 | 1.0237956 | 192.168.1.1 | 192.168.1.2 | ICMP | 78 | Echo (request) | 0.000000 | 0.000000 | 0.000000 |
| 4 | 2.0102232 | 192.168.1.2 | 192.168.1.1 | ICMP | 78 | Echo (reply) | 0.000000 | 0.000000 | 0.000000 |
| 5 | 2.0410448 | 192.168.1.1 | 192.168.1.2 | ICMP | 78 | Echo (request) | 0.000000 | 0.000000 | 0.000000 |
| 6 | 3.0128113 | 192.168.1.2 | 192.168.1.1 | ICMP | 78 | Echo (reply) | 0.000000 | 0.000000 | 0.000000 |
| 7 | 3.0268274 | 192.168.1.1 | 192.168.1.2 | ICMP | 78 | Echo (request) | 0.000000 | 0.000000 | 0.000000 |
| 8 | 4.0284444 | 192.168.1.2 | 192.168.1.1 | ICMP | 78 | Echo (reply) | 0.000000 | 0.000000 | 0.000000 |
| 9 | 4.0393392 | 192.168.1.1 | 192.168.1.2 | ICMP | 78 | Echo (request) | 0.000000 | 0.000000 | 0.000000 |
| 10 | 5.0134871 | 192.168.1.2 | 192.168.1.1 | ICMP | 78 | Echo (reply) | 0.000000 | 0.000000 | 0.000000 |
| 11 | 5.0217673 | 192.168.1.1 | 192.168.1.2 | ICMP | 78 | Echo (request) | 0.000000 | 0.000000 | 0.000000 |
| 12 | 5.0348275 | 192.168.1.2 | 192.168.1.1 | ICMP | 78 | Echo (reply) | 0.000000 | 0.000000 | 0.000000 |
| Total Delay | | | | | | | 0.000000 | 0.000000 | 0.000000 |
| Rata2 Delay | | | | | | | 0.000000 | 0.000000 | 0.000000 |

Gambar 4.16 Tampilan File csv Hasil Capture ICMP Ping ke-1 Redistribusi OSPF dan BGP.

Pada gambar 4.16 diatas, terdapat 2 tabel. Tabel yang disebelah kanan merupakan data yang ditangkap pada wireshark, sedangkan tabel yang berada bagian kiri merupakan proses pencarian nilai *delay*. Pada tabel proses pencarian nilai *delay*, terdapat kolom dengan keterangan Time 1, Time 2, Delay, Total Delay dan Rata2 Delay.

Time 1 didapatkan dari kolom Time pada tabel hasil *capture*. Karena pada penelitian ini berfokus pada paket ICMP, maka waktu yang diambil hanya waktu yang diperlukan oleh paket ICMP tersebut untuk melakukan *request* dan *reply* dari perangkat sumber menuju perangkat tujuan. Hal yang sama juga pada Time 2 didapatkan dari kolom Time pada tabel hasil *capture*. Perbedaan Time 1

dan Time 2 adalah proses pengambilan data waktu yang dilakukan pada Time 1 dimulai dari baris ke 3 (ICMP pertama) sampai baris 10 sedangkan pengambilan data waktu yang dilakukan pada Time 2 dimulai dari baris ke 4 (ICMP kedua) sampai baris 10.

Nilai dari kolom *delay* pada tabel yang dibawah didapatkan dari hasil pengurangan Time 2 dan Time 1. Total Delay merupakan jumlah dari seluruh *delay* yang terjadi pada kolom *delay* dan Rata - rata Delay didapatkan dari Total Delay dibagi dengan jumlah *bytes* paket yang dikirim 1 kali *ping*. Nilai dari Rata2 *delay* inilah yang diambil sebagai nilai *delay* pada saat proses 1 kali *ping* dilakukan. Dari gambar diatas, nilai *delay ping* ke-1 redistribusi BGP dan OSPF adalah sebesar 0,020194 *second* (s) atau 20,2 *millisecond* (ms).

Pada penelitian ini, proses *ping* dilakukan sebanyak 10 kali dimana 1 kali *ping* jumlah paket yang dikirim sebanyak 4 dan didapatkan nilai *delay* seperti keterangan yang ditampilkan tabel 4.3 berikut :

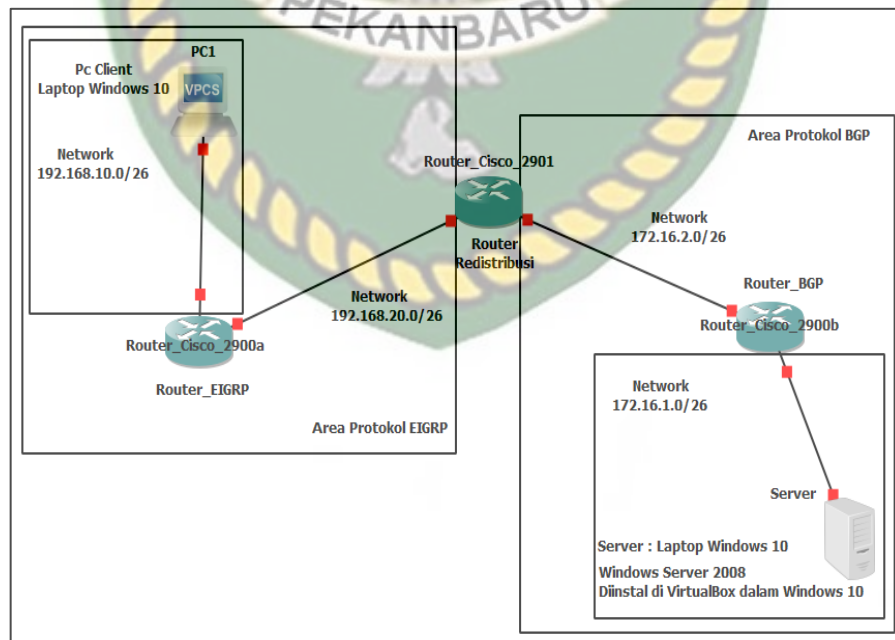
Tabel 4. 3 Nilai *Delay* Redistribusi OSPF dan BGP.

| NO. | Proses <i>Ping</i> | Nilai <i>Delay</i> (<i>millisecond</i> (ms)) |
|-----|--------------------|--|
| 1 | <i>Ping</i> ke -1 | 4,5 |
| 2 | <i>Ping</i> ke -2 | 4,4 |
| 3 | <i>Ping</i> ke -3 | 4,5 |
| 4 | <i>Ping</i> ke -4 | 4,4 |
| 5 | <i>Ping</i> ke -5 | 4,4 |
| 6 | <i>Ping</i> ke -6 | 20,2 |
| 7 | <i>Ping</i> ke -7 | 3,6 |
| 8 | <i>Ping</i> ke -8 | 3,6 |
| 9 | <i>Ping</i> ke -9 | 2,5 |
| 10 | <i>Ping</i> ke -10 | 3,6 |

4.4 Skema Analisa *Throughput*, *Packet Loss* dan *Delay* Redistribusi EIGRP dan BGP

Pada topologi jaringan BGP terdapat sebuah *server* dengan sistem operasi Windows Server 2008 yang diinstall dalam bentuk OS virtual dalam windows 10 dengan bantuan VirtualBox yang berfungsi sebagai perangkat tujuan dan di topologi BGP terdapat sebuah klien berupa laptop dengan sistem operasi Windows 10 sebagai perangkat sumber ketika melakukan *ping* untuk menganalisa nilai *throughput*, *packet loss* dan *delay* paket ICMP pada redistribusi OSPF dan BGP.

Aplikasi wireshark dipasang pada komputer yang ada pada jaringan EIGRP. Dengan menggunakan aplikasi ini peneliti mencari nilai *throughput*, *packet loss* dan *delay*. Gambar 4.17 berikut menampilkan topologi protokol EIGRP dan BGP yang didistribusi.



Gambar 4. 17 Topologi Jaringan Redistribusi EIGRP dan BGP.

Berdasarkan gambar 4.17 diatas, area EIGRP memiliki 2 *network* diantaranya *network* 192.168.10.0, *network* 192.168.20.0. Untuk area BGP juga memiliki 2 *network* diantaranya 172.16.1.0 dan 172.16.2.0.

4.4.1 Proses Analisa Nilai *Throughput* Redistribusi EIGRP dan BGP

Proses pencarian nilai *throughput* yang diterapkan adalah dengan melakukan penangkapan paket ICMP melalui *ping* pada (*command prompt*) cmd dengan bantuan aplikasi wireshark. Aplikasi wireshark dipasang pada komputer yang ada pada area EIGRP dan komputer ini bertindak sebagai perangkat sumber dalam melakukan *ping*.

Berdasarkan gambar 4.17, komputer ini diberi nama dengan PC1. Perangkat yang ada pada area EIGRP yang diberi nama Server digunakan sebagai perangkat tujuan. Pada penelitian ini perangkat sumber (PC1) diberi ip 192.168.10.2 dan perangkat tujuan (Server) diberi ip 172.16.1.4. Gambar 4.18 berikut menampilkan terkoneksiya perangkat sumber dan perangkat tujuan.

```

C:\Users\WAC&ping 172.16.1.4
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125

Ping statistics for 172.16.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 4ms, Average = 2ms

C:\Users\WAC&ping 192.16.1.4
Pinging 192.16.1.4 with 32 bytes of data:
Reply from 192.16.1.4: bytes=32 time=2ms TTL=125
Reply from 192.16.1.4: bytes=32 time=2ms TTL=125
Reply from 192.16.1.4: bytes=32 time=2ms TTL=125
Reply from 192.16.1.4: bytes=32 time=2ms TTL=125

Ping statistics for 192.16.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 2ms, Average = 1ms

C:\Users\WAC&ping 172.16.1.4
Pinging 172.16.1.4 with 32 bytes of data:
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125

Ping statistics for 172.16.1.4:

```

Gambar 4. 18 Koneksi Perangkat Sumber menuju Perangkat Tujuan.

Setelah perangkat sumber dan tujuan terkoneksi, langkah selanjutnya melakukan *ping* untuk mencari nilai *throughput*. Gambar 4.19 menampilkan proses *ping* dari perangkat sumber menuju perangkat tujuan (Server di BGP).

```

C:\Users\VACER>ping 172.16.1.4
Pinging 172.16.1.4 with 32 bytes of data:
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125

Ping statistics for 172.16.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\Users\VACER>ping 172.16.1.4
Pinging 172.16.1.4 with 32 bytes of data:
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125

Ping statistics for 172.16.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 2ms, Average = 2ms

C:\Users\VACER>ping 172.16.1.4
Pinging 172.16.1.4 with 32 bytes of data:
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125
Reply from 172.16.1.4: bytes=32 time=2ms TTL=125

Ping statistics for 172.16.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 2ms, Average = 2ms
  
```

Gambar 4. 19 Ping dari Laptop di EIGRP (Windows 10) menuju Server di BGP.

Pada saat bersamaan *ping* dilakukan, aplikasi wireshark aktif dan melakukan *capture* data paket ICMP. Batasan proses penangkapan data paket ICMP oleh wireshark adalah 1 kali *ping* dengan jumlah paket yang dikirim sebanyak 4. Gambar 4.20 berikut menampilkan penangkapan paket data ICMP oleh wireshark pada *ping* redistribusi EIGRP dan BGP.

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|-------------|-------------|----------|--------|---|
| 1 | 0.000000 | 192.168.1.2 | 192.168.1.4 | ICMP | 76 | 8080 (ping) request id=0x0000, seq=5752/34252, ttl=128 (reply in 2) |
| 2 | 0.002032 | 172.16.1.4 | 192.168.1.2 | ICMP | 76 | Echo (ping) reply id=0x0000, seq=5752/34252, ttl=125 (request in 1) |
| 3 | 0.012928 | 192.168.1.2 | 172.16.1.4 | ICMP | 80 | Device ID Router_EIGRP, Port ID 192.168.1.2/24/0 |
| 4 | 0.013024 | 192.168.1.2 | 172.16.1.4 | ICMP | 76 | Echo (ping) request id=0x0000, seq=5752/34252, ttl=128 (reply in 3) |
| 5 | 0.015024 | 172.16.1.4 | 192.168.1.2 | ICMP | 76 | Echo (ping) reply id=0x0000, seq=5752/34252, ttl=125 (request in 4) |
| 6 | 0.026976 | 192.168.1.2 | 192.168.1.4 | ICMP | 76 | 8080 (ping) request id=0x0000, seq=5752/34252, ttl=128 (reply in 2) |
| 7 | 0.031851 | 172.16.1.4 | 192.168.1.2 | ICMP | 76 | Echo (ping) reply id=0x0000, seq=5752/34252, ttl=125 (request in 5) |
| 8 | 0.051224 | 192.168.1.2 | 172.16.1.4 | ICMP | 76 | 8080 (ping) request id=0x0000, seq=5752/34252, ttl=128 (reply in 3) |
| 9 | 0.052224 | 172.16.1.4 | 192.168.1.2 | ICMP | 76 | Echo (ping) reply id=0x0000, seq=5752/34252, ttl=125 (request in 6) |
| 10 | 0.171222 | 192.168.1.2 | 172.16.1.4 | ICMP | 76 | Echo (ping) request id=0x0000, seq=5752/34252, ttl=128 (reply in 2) |

Gambar 4. 20 Penangkapan Paket ICMP oleh Wireshark pada Redistribusi EIGRP dan BGP.

Setelah melakukan pemberhentian *capture* data paket ICMP karena paket ICMP yang tercapture sudah memenuhi jumlah *request* dan *replay* 1 kali *ping*. Selanjutnya adalah menganalisa *capture file properties*. Gambar 4.21 berikut menampilkan hasil *capture file properties* pada redistribusi BGP dan EIGRP.



Gambar 4. 21 Tampilan *Capture File Properties* 1 kali *Ping* Pada Redistribusi EIGRP dan BGP.

Berdasarkan gambar 4.21 diatas, didapatkan nilai *throughput* 1 kali ping (*ping* ke-1) dengan jumlah paket yang dikirim sebanyak 4 sebesar 1219 bit/s.

Pada penelitian ini, proses *ping* dilakukan sebanyak 10 kali dimana 1 kali *ping* jumlah paket yang dikirim sebanyak 4 dan didapatkan nilai *throughput* seperti keterangan yang ditampilkan tabel 4.4 berikut :

Tabel 4. 4 Nilai *Throughput* Redistribusi BGP dan EIGRP.

| NO. | Proses <i>Ping</i> | Nilai <i>Throughput</i> (bytes/s) |
|-----|--------------------|-----------------------------------|
| 1 | <i>Ping</i> ke -1 | 1771 |
| 2 | <i>Ping</i> ke -2 | 2568 |
| 3 | <i>Ping</i> ke -3 | 1843 |
| 4 | <i>Ping</i> ke -4 | 1843 |
| 5 | <i>Ping</i> ke -5 | 2585 |
| 6 | <i>Ping</i> ke -6 | 1871 |
| 7 | <i>Ping</i> ke -7 | 1912 |
| 8 | <i>Ping</i> ke -8 | 1224 |

| | | |
|----|-------------|------|
| 9 | Ping ke -9 | 1230 |
| 10 | Ping ke -10 | 1383 |

4.4.2 Proses Analisa Nilai *Packet Loss* Redistribusi EIGRP dan BGP.

Proses mendapatkan nilai *packet loss* pada penelitian ini berdasarkan proses ping diambil dengan cara hasil tampilan *command prompt* (cmd) dan *capture file properties* milik wireshark. Gambar 4.22 menampilkan *packet loss* yang dihasilkan *ping* pada redistribusi EIGRP dan BGP pada cmd.



```

C:\Users\Riski Indra Saputra>ping 192.168.10.2

Pinging 192.168.10.2 with 32 bytes of data:
Reply from 192.168.10.2: bytes=32 time=45ms TTL=123
Reply from 192.168.10.2: bytes=32 time=21ms TTL=123
Reply from 192.168.10.2: bytes=32 time=18ms TTL=123
Reply from 192.168.10.2: bytes=32 time=17ms TTL=123

Ping statistics for 192.168.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 17ms, Maximum = 45ms, Average = 25ms

C:\Users\Riski Indra Saputra>ping 192.168.10.2

Pinging 192.168.10.2 with 32 bytes of data:
Reply from 192.168.10.2: bytes=32 time=16ms TTL=123
Reply from 192.168.10.2: bytes=32 time=17ms TTL=123
Reply from 192.168.10.2: bytes=32 time=18ms TTL=123
Reply from 192.168.10.2: bytes=32 time=21ms TTL=123

Ping statistics for 192.168.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 16ms, Maximum = 21ms, Average = 18ms

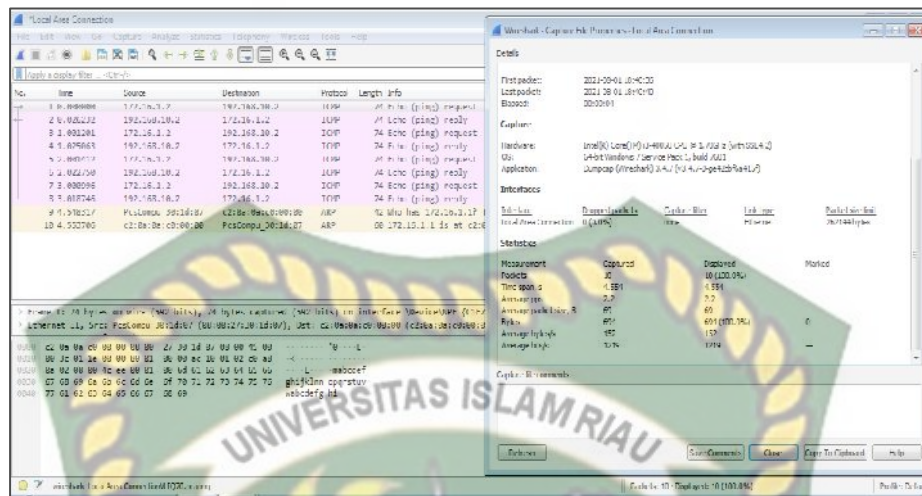
C:\Users\Riski Indra Saputra>

```

Gambar 4. 22 Tampilan *Packet Loss* yang dihasilkan *Ping* Pada Redistribusi BGP dan EIGRP pada cmd.

Berdasarkan gambar 4.22 diatas, *packet loss* yang didapat adalah 0 (0%).

Selanjutnya *packet loss* yang didapatkan dari hasil pada *capture file properties* pada wireshark dapat dilihat pada gambar 4.23 berikut ini.



Gambar 4. 23 Nilai *Packet Loss* dari Hasil *Capture File Properties* Pada Wireshark Redistribusi BGP dan EIGRP.

Berdasarkan gambar 4.23 diatas, nilai *packet loss* dapat diambil dari nilai *dropped packet*. Pada gambar terlihat nilai *dropped packet* adalah 0 (0%). Dari 2 cara ini dapat dilihat bahwa *packet loss* yang dihasilkan cmd dan *capture file properties* pada wireshark adalah sama. Berikut tabel hasil analisa *packet loss* yang dihasilkan 10 kali proses *ping* yang dilakukan pada penelitian ini.

Tabel 4. 5 Nilai *Packet Loss* Redistribusi EIGRP dan BGP.

| NO. | Proses <i>Ping</i> | Nilai <i>Packet Loss</i> (%) |
|-----|--------------------|---------------------------------|
| 1 | <i>Ping</i> ke -1 | 0 |
| 2 | <i>Ping</i> ke -2 | 0 |
| 3 | <i>Ping</i> ke -3 | 0 |
| 4 | <i>Ping</i> ke -4 | 0 |
| 5 | <i>Ping</i> ke -5 | 0 |
| 6 | <i>Ping</i> ke -6 | 0 |
| 7 | <i>Ping</i> ke -7 | 0 |
| 8 | <i>Ping</i> ke -8 | 0 |
| 9 | <i>Ping</i> ke -9 | 0 |
| 10 | <i>Ping</i> ke -10 | 0 |

4.4.3 Proses Analisa Nilai *Delay* Redistribusi BGP dan EIGRP

Proses pencarian nilai *delay* pada penelitian ini, dilakukan dengan melakukan penyimpanan file hasil *capture* pada wireshark ekstensi csv. Gambar 4.24 berikut menampilkan hasil *capture* paket ICMP dari proses *ping* ke-1 menggunakan wireshark pada redistribusi BGP dan EIGRP.

| No | Time | Source | Destination | Protocol | Length | Info |
|----|----------|-------------|-------------|----------|--------|--|
| 1 | 0.000000 | 192.168.1.2 | 192.168.1.4 | ICMP | 74 | Len=74 (ping) request id=60001, seq=2770/22296, ttl=120 (reply to 0) |
| 2 | 0.000000 | 192.168.1.4 | 192.168.1.2 | ICMP | 74 | Len=74 (ping) reply id=60001, seq=2770/22296, ttl=120 (request to 0) |
| 3 | 0.000000 | 192.168.1.2 | 192.168.1.4 | ICMP | 74 | Len=74 (ping) request id=60001, seq=2770/22296, ttl=120 (reply to 0) |
| 4 | 0.000000 | 192.168.1.4 | 192.168.1.2 | ICMP | 74 | Len=74 (ping) reply id=60001, seq=2770/22296, ttl=120 (request to 0) |
| 5 | 0.000000 | 192.168.1.2 | 192.168.1.4 | ICMP | 74 | Len=74 (ping) request id=60001, seq=2770/22296, ttl=120 (reply to 0) |
| 6 | 0.000000 | 192.168.1.4 | 192.168.1.2 | ICMP | 74 | Len=74 (ping) reply id=60001, seq=2770/22296, ttl=120 (request to 0) |
| 7 | 0.000000 | 192.168.1.2 | 192.168.1.4 | ICMP | 74 | Len=74 (ping) request id=60001, seq=2770/22296, ttl=120 (reply to 0) |
| 8 | 0.000000 | 192.168.1.4 | 192.168.1.2 | ICMP | 74 | Len=74 (ping) reply id=60001, seq=2770/22296, ttl=120 (request to 0) |
| 9 | 0.000000 | 192.168.1.2 | 192.168.1.4 | ICMP | 74 | Len=74 (ping) request id=60001, seq=2770/22296, ttl=120 (reply to 0) |
| 10 | 0.000000 | 192.168.1.4 | 192.168.1.2 | ICMP | 74 | Len=74 (ping) reply id=60001, seq=2770/22296, ttl=120 (request to 0) |
| 11 | 0.000000 | 192.168.1.2 | 192.168.1.4 | ICMP | 74 | Len=74 (ping) request id=60001, seq=2770/22296, ttl=120 (reply to 0) |
| 12 | 0.000000 | 192.168.1.4 | 192.168.1.2 | ICMP | 74 | Len=74 (ping) reply id=60001, seq=2770/22296, ttl=120 (request to 0) |
| 13 | 0.000000 | 192.168.1.2 | 192.168.1.4 | ICMP | 74 | Len=74 (ping) request id=60001, seq=2770/22296, ttl=120 (reply to 0) |

Gambar 4. 24 Tampilan Hasil Capture *Ping* Redistribusi EIGRP dan BGP.

Hasil dari *capture* tersebut disimpan menjadi file dengan ekstensi csv untuk melakukan perhitungan *delay* yang terjadi pada proses *ping* redistribusi EIGRP dan BGP. Tampilan file csv dari hasil *capture* paket ICMP *ping* ke-1 redistribusi OSPF dan BGP dapat dilihat pada gambar 4.25 berikut.

| No. | Time | Source | Destination | Protocol | Length | Info | Time 1 | Time 2 | Delay |
|-----|----------|---------------|---------------|----------|--------|--|-------------|----------|------------|
| 1 | 0 | 192.168.10.2 | 172.16.1.4 | ICMP | 74 | 0x0001, seq=9797/24952, ttl=128 (reply in 2) | 0,002772 | 0 | 0,002772 |
| 2 | 0,002772 | 172.16.1.4 | 192.168.10.2 | ICMP | 74 | 0x0001, seq=7697/30482, ttl=128 (request in 1) | 0,179458 | 0,002772 | 0,179664 |
| 3 | 0,179458 | 192.168.10.2 | 172.16.1.4 | ICMP | 74 | 0x0001, seq=9797/24952, ttl=128 (reply in 3) | 1,013935 | 0,179458 | 0,834477 |
| 4 | 1,013935 | 192.168.10.2 | 172.16.1.4 | ICMP | 74 | 0x0001, seq=9797/24952, ttl=128 (reply in 5) | 1,013935 | 1,013935 | 0,834477 |
| 5 | 1,013935 | 192.168.10.2 | 172.16.1.4 | ICMP | 74 | 0x0001, seq=9797/24952, ttl=128 (reply in 7) | 2,029485 | 1,013935 | 1,015550 |
| 6 | 2,029485 | 192.168.10.2 | 172.16.1.4 | ICMP | 74 | 0x0001, seq=9797/24952, ttl=128 (reply in 9) | 3,045035 | 2,029485 | 1,015550 |
| 7 | 3,045035 | 192.168.10.2 | 172.16.1.4 | ICMP | 74 | 0x0001, seq=9797/24952, ttl=128 (reply in 11) | 4,060585 | 3,045035 | 1,015550 |
| 8 | 3,045035 | 192.168.10.2 | 172.16.1.4 | ICMP | 74 | 0x0001, seq=9797/24952, ttl=128 (reply in 13) | 5,076135 | 4,060585 | 1,015550 |
| 9 | 3,045035 | 192.168.10.2 | 172.16.1.4 | ICMP | 74 | 0x0001, seq=9797/24952, ttl=128 (reply in 15) | 6,091685 | 5,076135 | 1,015550 |
| 10 | 3,172682 | 192.168.10.1 | 224.0.0.10 | ICMP | 74 | 0x0001, seq=9797/24952, ttl=128 (reply in 17) | 7,107235 | 6,091685 | 1,015550 |
| 11 | 4,578363 | Cisco_64b703a | Cisco_4f13c9c | ARP | 47 | 32.168.10.17 Full 32.168.10.2 | | | |
| 12 | 4,578363 | Cisco_0134693 | Cisco_3a07706 | ARP | 50 | 10.1.1.44 44.44.44.44:50 | | | |
| 13 | 5,120279 | Cisco_0134693 | Cisco_4f13c9c | ICMP | 90 | Reply | | | |
| | | | | | | | Total Delay | | 3,052335 |
| | | | | | | | Rate2 Delay | | 0,00253187 |

Gambar 4. 25 Tampilan File csv Hasil Capture ICMP Ping Redistribusi EIGRP dan BGP.

Pada gambar 4.25 diatas, terdapat 2 tabel. Tabel yang di sebelah kiri merupakan data yang dicapture pada wireshark, sedangkan tabel yang di sebelah kanan merupakan proses pencarian nilai delay. Pada tabel proses pencarian nilai delay, terdapat kolom dengan keterangan Time 1, Time 2, Delay, Total Delay dan Rata2 Delay.

Time 1 didapatkan dari kolom Time pada tabel hasil *capture*. Karena pada penelitian ini berfokus pada paket ICMP, maka waktu yang diambil hanya waktu yang diperlukan oleh paket ICMP tersebut untuk melakukan *request* dan *reply* dari perangkat sumber menuju perangkat tujuan. Hal yang sama juga pada Time 2 didapatkan dari kolom Time pada tabel hasil *capture*. Perbedaan Time 1 dan Time 2 adalah proses pengambilan data waktu yang dilakukan pada Time 1 dimulai dari baris ke 1 (ICMP pertama) sampai baris 10 sedangkan pengambilan data waktu yang dilakukan pada Time 2 dimulai dari baris ke 2 (ICMP kedua) sampai baris 10.

Nilai dari kolom *delay* pada tabel yang dibawah didapatkan dari hasil pengurangan Time 2 dan Time 1. Total Delay merupakan jumlah dari seluruh nilai *delay* yang ada pada kolom *delay* dan Rata2 Delay didapatkan dari Total Delay dibagi dengan jumlah *bytes* paket yang dikirim 1 kali *ping*. Nilai dari Rata2 *delay* inilah yang diambil sebagai nilai *delay* pada saat proses 1 kali *ping* dilakukan. Dari gambar diatas, nilai *delay ping* redistribusi EIGRP dan BGP adalah sebesar 0,00255287 *second* atau 2,5 *millisecond* (ms).

Pada penelitian ini, proses *ping* dilakukan sebanyak 10 kali dimana 1 kali *ping* jumlah paket yang dikirim sebanyak 4 dan didapatkan nilai *delay* seperti keterangan yang ditampilkan tabel 4.6 berikut :

Tabel 4. 6 Nilai *Delay* Redistribusi BGP dan EIGRP.

| NO. | Proses <i>Ping</i> | Nilai <i>Delay</i> (<i>millisecond</i> (ms)) |
|-----|--------------------|--|
| 1 | <i>Ping</i> ke -1 | 4,3 |
| 2 | <i>Ping</i> ke -2 | 2,9 |
| 3 | <i>Ping</i> ke -3 | 4,3 |
| 4 | <i>Ping</i> ke -4 | 4,3 |
| 5 | <i>Ping</i> ke -5 | 2,9 |
| 6 | <i>Ping</i> ke -6 | 2,5 |
| 7 | <i>Ping</i> ke -7 | 4,2 |
| 8 | <i>Ping</i> ke -8 | 3,4 |
| 9 | <i>Ping</i> ke -9 | 3,8 |
| 10 | <i>Ping</i> ke -10 | 3,4 |

4.5 Perbandingan Nilai *Throughput*, *Packet Loss*, dan *Delay* antara Redistribusi BGP dan OSPF dengan Redistribusi BGP dan EIGRP

Pada penelitian ini, peneliti menggunakan nilai *Throughput*, *Packet Loss* dan *Delay* sebagai tolak ukur untuk membandingkan dari kinerja protokol jaringan yang menerapkan metode redistribusi. Pada penelitian ini, redistribusi protokol

jaringan yang dibandingkan adalah redistribusi OSPF dan BGP dengan redistribusi EIGRP dan BGP.

4.5.1 Perbandingan Nilai *Throughput* antara Redistribusi OSPF dan BGP dengan Redistribusi EIGRP dan BGP.

Berdasarkan data nilai *throughput* pada masing masing redistribusi protokol jaringan yang dilakukan, telah didapatkan data seperti yang ditampilkan pada tabel 4.7 berikut.

Tabel 4. 7 Perbandingan Nilai *Throughput* antara Redistribusi BGP dan OSPF dengan Redistribusi BGP dan EIGRP.

| No. | Proses <i>Ping</i> | <i>Throughput (bytes/s)</i> | |
|--|---------------------|-----------------------------|----------------------------|
| | | Redistribusi OSPF dan BGP | Redistribusi EIGRP dan BGP |
| 1 | <i>Ping</i> ke – 1 | 1771 | 1219 |
| 2 | <i>Ping</i> ke – 2 | 1818 | 2568 |
| 3 | <i>Ping</i> ke – 3 | 1777 | 1843 |
| 4 | <i>Ping</i> ke – 4 | 1825 | 1843 |
| 5 | <i>Ping</i> ke – 5 | 1835 | 2585 |
| 6 | <i>Ping</i> ke – 6 | 1219 | 1871 |
| 7 | <i>Ping</i> ke – 7 | 1077 | 1912 |
| 8 | <i>Ping</i> ke – 8 | 1094 | 1224 |
| 9 | <i>Ping</i> ke – 9 | 1343 | 1230 |
| 10 | <i>Ping</i> ke – 10 | 1438 | 1383 |
| Total <i>Throughput (bytes/s)</i> | | 15197 | 17678 |

Berdasarkan tabel 4.7 diatas, terlihat total nilai *throughput* redistribusi EIGRP dan BGP sebesar 17678 *bytes per second* sedangkan nilai *throughput* redistribusi OSPF dan BGP sebesar 15197 *bytes per second* . terlihat bahwa nilai *throughput* redistribusi EIGRP dan BGP lebih besar dibandingkan redistribusi OSPF dan BGP.

4.5.2 Perbandingan Nilai *Packet Loss* antara Redistribusi BGP dan OSPF dengan Redistribusi BGP dan EIGRP

Berdasarkan data nilai *packet loss* pada masing masing redistribusi protokol jaringan yang dilakukan, telah didapatkan data seperti yang ditampilkan pada tabel 4.8 berikut.

Tabel 4. 8 Perbandingan Nilai *Packet Loss* antara Redistribusi OSPF dan BGP dengan Redistribusi EIGRP dan BGP.

| No. | Proses <i>Ping</i> | <i>Packet Loss</i> (%) | |
|-------------------------------|---------------------|---------------------------|----------------------------|
| | | Redistribusi BGP dan OSPF | Redistribusi BGP dan EIGRP |
| 1 | <i>Ping</i> ke – 1 | 0 | 0 |
| 2 | <i>Ping</i> ke – 2 | 0 | 0 |
| 3 | <i>Ping</i> ke – 3 | 0 | 0 |
| 4 | <i>Ping</i> ke – 4 | 0 | 0 |
| 5 | <i>Ping</i> ke – 5 | 0 | 0 |
| 6 | <i>Ping</i> ke – 6 | 0 | 0 |
| 7 | <i>Ping</i> ke – 7 | 0 | 0 |
| 8 | <i>Ping</i> ke – 8 | 0 | 0 |
| 9 | <i>Ping</i> ke – 9 | 0 | 0 |
| 10 | <i>Ping</i> ke – 10 | 0 | 0 |
| Total Packets Loss (%) | | 0 | 0 |

Berdasarkan tabel 4.8 diatas, terlihat total nilai *packet loss* redistribusi EIGRP dan BGP memiliki nilai yang sama dengan redistribusi OSPF dan BGP, yaitu sebesar 0%. Dari data dari tabel 4.8 diatas, memberikan informasi bahwa setiap *ping* yang dilakukan sebanyak 10 kali pada penelitian ini paket yang dikirim selalu berhasil dikirim dan diterima oleh perangkat sumber ke perangkat tujuan.

4.5.3 Perbandingan Nilai *Delay* antara Redistribusi OSPF dan BGP dengan Redistribusi EIGRP dan BGP.

Berdasarkan data nilai *delay* pada masing masing redistribusi protokol jaringan yang dilakukan, telah didapatkan data seperti yang ditampilkan pada tabel 4.9 berikut.

Tabel 4. 9 Perbandingan Nilai *Delay* antara Redistribusi OSPF dan BGP dengan Redistribusi EIGRP dan BGP.

| No. | Proses <i>Ping</i> | <i>Packet Delay</i> (ms) | |
|-------------------------------|---------------------|---------------------------|----------------------------|
| | | Redistribusi OSPF dan BGP | Redistribusi EIGRP dan BGP |
| 1 | <i>Ping</i> ke – 1 | 4,5 | 4,3 |
| 2 | <i>Ping</i> ke – 2 | 4,4 | 2,9 |
| 3 | <i>Ping</i> ke – 3 | 4,5 | 4,3 |
| 4 | <i>Ping</i> ke – 4 | 4,4 | 4,3 |
| 5 | <i>Ping</i> ke – 5 | 4,3 | 2,9 |
| 6 | <i>Ping</i> ke – 6 | 20,2 | 2,5 |
| 7 | <i>Ping</i> ke – 7 | 3,6 | 4,2 |
| 8 | <i>Ping</i> ke – 8 | 3,6 | 3,4 |
| 9 | <i>Ping</i> ke – 9 | 2,5 | 3,8 |
| 10 | <i>Ping</i> ke – 10 | 3,6 | 3,4 |
| Total <i>Delay</i> (s) | | 55,6 | 36 |

Berdasarkan tabel 4.9 diatas, terlihat total nilai *delay* redistribusi EIGRP dan BGP lebih kecil dibandingkan redistribusi OSPF dan BGP.

4.6 Kesimpulan Analisa Kinerja Redistribusi BGP dan OSPF dengan Redistribusi BGP dan EIGRP Berdasarkan Nilai *Throughput*, *Packet Loss* dan *Delay*

Nilai *Packet Loss* redistribusi BGP dengan OSPF dan redistribusi BGP dengan EIGRP memiliki nilai yang sama,itu artinya setiap *ping* yang dilakukan paket yang dikirim selalu berhasil. Gambar 4.26 dibawah ini menampilkan

perbandingan nilai *packet loss* antara redistribusi OSPF dan BGP dengan EIGRP dan BGP.

| No. | Proses Ping | Packet Loss (%) | |
|-------------------------------|--------------|---------------------------|----------------------------|
| | | Redistribusi BGP dan OSPF | Redistribusi BGP dan EIGRP |
| 1 | Ping ke - 1 | 0 | 0 |
| 2 | Ping ke - 2 | 0 | 0 |
| 3 | Ping ke - 3 | 0 | 0 |
| 4 | Ping ke - 4 | 0 | 0 |
| 5 | Ping ke - 5 | 0 | 0 |
| 6 | Ping ke - 6 | 0 | 0 |
| 7 | Ping ke - 7 | 0 | 0 |
| 8 | Ping ke - 8 | 0 | 0 |
| 9 | Ping ke - 9 | 0 | 0 |
| 10 | Ping ke - 10 | 0 | 0 |
| Total Packets Loss (%) | | 0 | 0 |

Gambar 4. 26 Perbandingan Nilai *Packet Loss*.

Tetapi, walaupun memiliki nilai *packet Loss* yang sama nilai *throughput* dan *delay* berbeda. Gambar 4.27 dibawah ini menampilkan perbandingan nilai *throughput* antara redistribusi OSPF dan BGP dengan EIGRP dan BGP.

| No. | Proses Ping | Throughput (bytes/s) | |
|-----------------------------------|--------------|---------------------------|----------------------------|
| | | Redistribusi OSPF dan BGP | Redistribusi EIGRP dan BGP |
| 1 | Ping ke - 1 | 1771 | 1219 |
| 2 | Ping ke - 2 | 1818 | 2568 |
| 3 | Ping ke - 3 | 1777 | 1843 |
| 4 | Ping ke - 4 | 1825 | 1843 |
| 5 | Ping ke - 5 | 1835 | 2585 |
| 6 | Ping ke - 6 | 1219 | 1871 |
| 7 | Ping ke - 7 | 1077 | 1912 |
| 8 | Ping ke - 8 | 1094 | 1224 |
| 9 | Ping ke - 9 | 1343 | 1230 |
| 10 | Ping ke - 10 | 1438 | 1383 |
| Total Throughput (bytes/s) | | 15197 | 17678 |

Gambar 4. 27 Perbandingan Nilai *Throughput*.

Dari gambar 4.27 diatas, nilai *throughput* redistribusi BGP dengan OSPF lebih kecil dibandingkan nilai *throughput* redistribusi BGP dengan EIGRP. Tetapi sebaliknya, nilai *delay* redistribusi BGP dengan OSPF lebih besar dibandingkan nilai *delay* redistribusi BGP dengan EIGRP. Gambar 4.28 dibawah ini

menampilkan perbandingan nilai *delay* antara redistribusi OSPF dan BGP dengan EIGRP dan BGP.

| No. | Proses Ping | Packet Delay (ms) | |
|------------------------|--------------|---------------------------|----------------------------|
| | | Redistribusi OSPF dan BGP | Redistribusi EIGRP dan BGP |
| 1 | Ping ke - 1 | 4,5 | 4,3 |
| 2 | Ping ke - 2 | 4,4 | 2,9 |
| 3 | Ping ke - 3 | 4,5 | 4,3 |
| 4 | Ping ke - 4 | 4,4 | 4,3 |
| 5 | Ping ke - 5 | 4,3 | 2,9 |
| 6 | Ping ke - 6 | 20,2 | 2,5 |
| 7 | Ping ke - 7 | 3,6 | 4,2 |
| 8 | Ping ke - 8 | 3,6 | 3,4 |
| 9 | Ping ke - 9 | 2,5 | 3,8 |
| 10 | Ping ke - 10 | 3,6 | 3,4 |
| Total Delay (s) | | 55,6 | 36 |

Gambar 4. 28 Perbandingan Nilai Delay.

Dari hasil data tersebut peneliti juga menyimpulkan bahwa nilai *throughput* pada penelitian ini berbanding terbalik dengan nilai *delay*. Jika nilai *throughput* besar maka nilai *delay* akan kecil. Begitu juga sebaliknya, jika nilai *throughput* kecil maka nilai *delay* akan besar.

Jika dilihat dari nilai *delay*, dapat disimpulkan bahwa redistribusi EIGRP dan BGP (36 ms) lebih efisien dari segi waktu dibandingkan redistribusi OSPF dan BGP (55,6 ms), karena nilainya waktu *delay*nya lebih kecil. Selisih waktu kedua redistribusi itu adalah 19,6 ms dan hal itu sangat berpengaruh pada komunikasi data dalam jaringan . Selain itu, pada saat proses penangkapan paket ICMP menggunakan wireshark terjadi perbedaan paket komunikasi antara EIGRP dan OSPF. Gambar 4.29 berikut menampilkan hasil penangkapan paket komunikasi oleh wireshark antara redistribusi OSPF dan BGP.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan pembahasan yang telah dipaparkan pada penelitian dengan judul Analisis Kinerja *Redistribution Routing* Protokol OSPF, EIGRP dan BGP, dapat ditarik beberapa kesimpulan sebagai berikut, :

1. Redistribusi antara OSPF dengan BGP memiliki nilai *throughput* lebih kecil dibandingkan nilai *throughput* redistribusi antara EIGRP dengan BGP, dimana nilai total *throughput* Redistribusi antara OSPF dengan BGP sebesar 15197 *bytes per second*, sedangkan nilai *throughput* redistribusi antara EIGRP dengan BGP sebesar 17678 *bytes per second*.
2. Redistribusi antara OSPF dengan BGP memiliki nilai *delay* lebih besar dibandingkan nilai *delay* redistribusi antara EIGRP dengan BGP, dimana nilai total *delay* redistribusi antara OSPF dengan BGP sebesar 0,055769 *second*, sedangkan nilai *delay* redistribusi antara EIGRP dan BGP sebesar 0,036133 *second*.
3. Nilai *packet loss* yang terjadi pada penelitian ini, redistribusi antara BGP dengan OSPF dan redistribusi antara BGP dengan EIGRP adalah sama yaitu 0%, walaupun demikian kedua redistribusi yang diterapkan pada penelitian ini memiliki nilai *delay* dan *throughput* yang berbeda.

4. Berdasarkan penelitian ini, penulis menyimpulkan bahwa nilai *delay* dan *throughput* berbanding terbalik, artinya jika nilai *throughput* suatu komunikasi jaringan besar, maka nilai delaynya akan kecil. Sebaliknya, jika nilai *throughput* suatu komunikasi jaringan kecil, maka nilai delaynya akan besar. Dengan demikian penulis menyimpulkan bahwa redistribusi BGP dan EIGRP lebih baik dibandingkan dengan redistribusi BGP dengan OSPF karena memiliki *delay* yang kecil.
5. Menurut peneliti, redistribusi BGP dengan EIGRP lebih baik dibandingkan redistribusi BGP dengan OSPF. hal ini berdasarkan nilai *delay* redistribusi BGP dengan EIGRP relative kecil (36 ms) dibandingkan redistribusi BGP dengan OSPF (55,6 ms).
6. Terdapat perbedaan cara komunikasi antara redistribusi OSPF dan BGP dengan redistribusi EIGRP dan BGP. Pada redistribusi OSPF dan BGP terdapat paket yang ditangkap wireshark selain ICMP yaitu dengan nama *Hello Packets* dengan *length* 90 dan pada redistribusi EIGRP dan BGP paket itu adalah *Hello* dengan *length* 74 dan menurut penulis hal ini mempengaruhi nilai *delay* dari kedua redistribusi protokol tersebut.

5.2 Saran

Simulasi pada penelitian ini masih sederhana dan bisa terjadi pengembangan lagi. Adapun saran dari penulis untuk pengembangan lebih lanjut diantaranya, :

1. Pada penelitian ini, paket yang ditangkap hanya ICMP melalui *ping* pada cmd sehingga bisa saja berubah hasilnya jika untuk paket yang lebih kompleks, misal paket SMTP, HTTP, HTTPS.
2. Jaringan yang disimulasikan dalam bentuk lokal, dimana konfigurasi ip dan protokol *routing* dinamis dilakukan secara manual, untuk hasil yang lebih kompleks bisa dilakukan dengan menghubungkan dengan jaringan internet.



DAFTAR PUSTAKA

- A. Forouzan, behrouze dan chung fegan sophia.(2007). *Data Communication and networking, fourth edition USA : Prentice Hall Inc.*
- Al Ghivani, Akhmad Zulfikar “ Perbandingan *Routing* Protokol BGP dan EIGRP, Evaluasi Kinerja Performansi pada *Autonomous System* Berbeda” , Jurnal SISTEMASI, Volume 7, Nomor 2, Mei 2018 : 95 – 105.
- Al-Musawi, B., Branch, P., & Armitage, G. (2017). *BGP anomaly detection techniques: A survey. IEEE Communications Surveys & Tutorials*, 19(1), 377-396.
- A,O'brien , James, 2006, *Introducing To Information System*, Jakarta: Salemba Empat,.
- Aryanta, D., & Pranata, B. A. (2014). Perancangan dan Analisis Redistribusion Routing Protocol OSPF dan EIGRP. *ELKOMIKA*, 2(2), 85–99.
- Athira, M., Abrahami, L., & Sangeetha, R. G. (2017, March). *Study on network performance of interior gateway protocols—RIP, EIGRP and OSPF. In Nextgen Electronic Technologies: Silicon to Software (ICNETS2)*, 2017 *International Conference on* (pp. 344-348). IEEE.
- Atif Manzoor, Muzammil Hussain1 dan Sobia Mehrban “*Performance Analysis and Route Optimization: Redistribution between EIGRP, OSPF & BGP Routing*

Protocols ; Journal Pre-proof, PII : S0920-5489(19)30099-6, Reference: CSI 103391, University of Management and Technology, Lahore, Pakistan (2019).

Circiumarescu, L. D., Predusca, G., Angelescu, N., & Puchianu, D. (2015, May).

Comparative Analysis of Protocol RIP, OSPF, RIGRP and IGRP for Service Video Conferencing, E-mail, FTP, HTTP. In Control Systems and Computer Science (CSCS), 2015 20th International Conference on (pp. 584-589). IEEE.

Charles M. Kozierok (20 September 2005), "Overview of FTP Operation,"FTP

Overview, History and Standards, dari tcp ip guide.

URL:http://www.tcpipguide.com/free/t_FTPOverviewHistoryandStandards-2.html.

Darmawan, Teguh Imanto "Analisa *Link Balancing* dan *Failover 2 Provider*

Menggunakan *Border Gateway Protocol (BGP)* Pada Router Cisco 7606s ", Jurnal Nasional Teknologi dan Sistem Informasi, Vol 03 No. 03 (2017) 326 – 333, Jurusan Teknik Elektro Fakultas Teknik Universitas Andalas, Kampus Unand Limau Manis, Padang, 25163, Indonesia.

Dumitrache, C. G., Predusca, G., Circiumarescu, L. D., Angelescu, N., & Puchianu,

D. C. (2017, October). *Comparative study of RIP, OSPF and EIGRP protocols using Cisco Packet Tracer*. In Electrical and Electronics Engineering (ISEEE), 2017 5th *International Symposium* on (pp. 1-6). IEEE.

Dwi Aryanta dan Bayu Agung Pranata “ Perancangan dan Analisis *Redistribution Routing Protocol* OSPF dan EIGRP ”, Teknik Elektro Itenas | No.2 | Vol. 2, Institut Teknologi Nasional Juli - Desember 2014.

Erma Julita, Aidil Halim Lubis dan Muhammad Zarlis, (2017). “ Analisis *Routing EIGRP* dalam Menentukan *Router* yang dilalui pada WAN ”. Jurnal & Penelitian Teknik Informatika, Volume 1, Nomor 2, April 2017. E - ISSN: 2541 2019 p-ISSN : 2541-044X 23.

Hasanah, F. U., & Mubarakah, N. (2014). Analisis Kinerja *Routing* Dinamis Dengan Teknik Rip (*Routing Information Protocol*) Pada Topologi Ring Dalam Jaringan Lan (*Local Area Network*) Menggunakan Cisco Packet Tracer. *SINGUDA ENSIKOM*, 7(Juni), 118–124.

Hikmaturokhman, A., Purwanto, A., & Munadi, R. (2010). Analisis Perancangan Dan Implementasi *Firewall* Dan *Traffic Filtering* Menggunakan Cisco Router. *Seminar Nasional Informatika*, 2010 (semnasIF), C1–C8.

Maulana, A. (2018). Konsep dan Penerapan Routing OSPF, dan EIGRP dengan Metode Redistribution. *AKRAB JUARA*, 3 (February), 154.

Masykur, F.(2016). ”*PENGGABUNGAN ANTAR ROUTING PROTOCOL MENGGUNAKAN TEKNIK REDISTRIBUTION*”. ISBN: 978-602-1180-33-4 *Prosiding SNATI F Ke - 3 Tahun 2016*, 39-42.

Musril, H. A. (2015). ANALISIS UNJUK KERJA RIPv2 DAN EIGRP DALAM DYNAMIC ROUTING PROTOCOL THE PERFORMANCE ANALYSIS

OF RIPv2 AND EIGRP ON. *Jurnal Elektro Telekomunikasi Terapan*,
1(Desember), 116–124.

Musril, H. A. (2017). Simulasi Interkoneksi Antara Autonomous System (As)
Menggunakan Border Gateway Protocol (Bgp). *Jurnal Nasional Informatika
Dan Teknologi Jaringan E-ISSN:2540-7600 P-ISSN : 2540-7597, Vol 2, No
1(26181)*, 1–9.

mrizqiariadi. 2014. Pengertian Routing , Tabel *Routing* & Protokol *Routing*.
<https://mrizqiariadi.wordpress.com>. [Online] 6 22, 2014. [Cited: 8 1, 2017.]
[https://mrizqiariadi.wordpress.com/2014/06/22/pengertian-routing-tabel-
routingprotokol-routing/](https://mrizqiariadi.wordpress.com/2014/06/22/pengertian-routing-tabel-routingprotokol-routing/).

Nazir, Moh. (2013). Metode Penelitian. Bogor : *Ghalia Indonesia*.

Priyambodho, Dimas. (2014). Analisis Kinerja *Routing* Protokol EIGRP dan OSPF
pada Topologi *Ring* dan *Mesh*. Bandung : Institut Teknologi Nasional Bandung

Sirat Muhidin. (2006). Penelitian Tindak Kelas. Jakarta: *Gaung Persada*.

Sofana, 2008, " Membangun Jaringan Komputer", bandung: Informatika,.

Sofana, Iwan. (2012). CISCO CCNP dan jaringan Komputer (Materi *Route, Switch,
& Troubleshooting*). Bandung : Informatika.

Sofana, Iwan. 2013. Membangun Jaringan Komputer : Mudah membuat Jaringan
Komputer (Wire & Wireless) untuk pengguna Windows dan Linux. Bandung:
Informatika.

Villasica, Yovie Dwi dan Mubarakah Naemah, (2014), Analisis Kinerja Routing
Dinamis Dengan Teknik Osp (*Open Shortest Path First*) Pada Topologi Mesh

Dalam Jaringan *Local Area Network* (Lan) Menggunakan Cisco Packet Tracer,
SINGUDA ENSIKOM VOL. 7 NO. 3 Juni 2014.

Winarno, Surakhmad. (2004). Pengantar Penelitian Ilmiah, Dasar, Metode, dan Teknik. Bandung : *Tarsito.*

Wood, T., Ramakrishnan, K. K., Hwang, J., Liu, G., & Zhang, W. (2015). Toward a software-based network: integrating software defined networking and network function virtualization. *IEEE Network*, 29(3), 36-41.

Yoldi Novendra, Yudhi Arta dan Apri Siswanto “Analisis Kinerja Routing OSPF dan EIGRP” , *IT Journal Research and Development* Vol.2, No.2, Maret 2018.

