UNIVERSITAS INDONESIA

PERBANDINGAN PERFORMANSI JARINGAN IPv6 *TUNNELING* GRE DAN ISATAP PADA APLIKASI FTP

SKRIPSI

GHIFFARI AULIA
0806339105

FAKULTAS TEKNIK
DEPARTEMEN TEKNIK ELEKTRO
PROGRAM STUDI TEKNIK KOMPUTER
DEPOK
DESEMBER 2012

UNIVERSITAS INDONESIA

PERBANDINGAN PERFORMANSI JARINGAN IPv6

TUNNELING GRE DAN ISATAP APLIKASI FTP

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

GHIFFARI AULIA
0806339105

FAKULTAS TEKNIK
DEPARTEMEN TEKNIK ELEKTRO
PROGRAM STUDI TEKNIK KOMPUTER
DEPOK
DESEMBER 2012

HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama	**: Ghiffari Aulia**
NPM	**: 0806339105**
Tanda Tangan	**: .....**
Tanggal	**: 28 Desember 2012**

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama : Ghiffari Aulia

NPM : 0806339105

Program Studi : Teknik Komputer

Judul Skripsi : **PERBANDINGAN PERFORMANSI JARINGAN IPv6 *TUNNELING* GRE DAN ISATAP PADA APLIKASI FTP**

Telah berhasil dipertahankan di hadapan penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Komputer, Fakultas Teknik, Universitas Indonesia.

PENGUJI

Pembimbing : Ir. Endang Sriningsih, MT (.....)

Penguji 1 : Dr. Ir. Anak Agung Putri Ratna, M.Eng (.....)

Penguji 2 : Prima Dewi Purnamasari, ST, M.Sc (.....)

Ditetapkan di : Depok

Tanggal : 21 Januari 2013

KATA PENGANTAR

Puji serta syukur penulis panjatkan kehadirat Allah SWT karena dengan limpahan berkat, rahmat serta hidayat-Nya penulis dapat menyelesaikan skripsi ini. Penulisan skripsi dilakukan sebagai salah satu syarat untuk menjadi Sarjana Teknik di Departemen Teknik Elektro FTUI. Saya menyadari tanpa bantuan banyak pihak, skripsi ini tidak mungkin terselesaikan. Oleh sebab itu, pada kesempatan ini saya ingin mengucapkan terima kasih kepada pihak-pihak yang membantu saya dalam segala hal mengenai penyusunan skripsi baik secara langsung maupun tidak langsung. Hal ini penulis ditujukan kepada:

1. Allah SWT yang telah memberikan kekuatan kepada penulis untuk menyelesaikan skripsi ini.
2. Orang tua dan keluarga penulis.
3. Ir. Endang Sriningsih, MT selaku dosen pembimbing skripsi yang telah meluangkan waktunya, serta masukan-masukan selama bimbingan.
4. Muhammad Salman, ST, MIT atas bimbingannya selama ini sebagai kepala laboratorium jaringan.
5. Rekan-rekan asisten laboratorium jaringan terutama Lina Afriana sehingga saya bisa menggunakan *server* yang ada di laboratorium.
6. Muhammad Normansyah Pnd, selaku teman berdiskusi selama penyusunan skripsi.

Akhir kata, penulis berharap agar Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga laporan skripsi ini membawa manfaat.

Depok, 28 Desember 2012

Penulis

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika Universitas Indonesia, saya yang bertanda tangan di bawah ini :

Nama	: Ghiffari Aulia
NPM	: 0806339105
Program Studi	: Teknik Komputer
Departemen	: Teknik Elektro
Fakultas	: Teknik
Jenis Karya	: Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right)*** atas karya ilmiah saya yang berjudul :

PERBANDINGAN PERFORMANSI JARINGAN IPv6 *TUNNELING* GRE DAN ISATAP PADA APLIKASI FTP

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/ formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Tanggal : 28 Desember 2012

Yang menyatakan

(Ghiffari Aulia)

ABSTRAK

Nama : Ghiffari Aulia
Program Studi : Teknik Komputer
Judul : Perbandingan Performansi Jaringan IPv6 *Tunneling* GRE dan ISATAP pada aplikasi FTP

Pada skripsi ini akan dibangun suatu jaringan sederhana untuk mengamati performansi aplikasi transfer file pada jaringan IPv6 tunneling GRE dan ISATAP beserta perbandingannya. Tunneling IPv6 adalah fitur pada jaringan IPv6 untuk membantu migrasi jaringan IPv4 ke IPv6 secara bertahap. Pada proses integrasi ke jaringan tunneling, pemilihan tipe tunneling harus berdasarkan aplikasi yang dijalankan. Penulisan skripsi ini bertujuan untuk membandingkan kedua tipe tunneling (GRE dan ISATAP) pada FTP berdasarkan parameter-parameter QoS. Parameter-parameter QoS yang dibandingkan merupakan parameter penting untuk menentukan *tunneling* mana yang lebih baik dalam mengantarkan paket TCP.

Throughput pada jaringan *tunneling* ISATAP mengalami kenaikan sebesar 0,15% dari *throughput* pada jaringan *tunneling* GRE. *Delay* juga tidak banyak berbeda, pada *tunneling* ISATAP *delay* hanya menurun sebesar 0,98%. Perbedaan yang signifikan terjadi pada *packet loss* dimana ISATAP mempunyai *packet loss* yang lebih besar, yaitu 7,488% dibandingkan *packet loss* pada GRE yang bernilai 5,562%. Oleh karena itu, *tunneling* GRE lebih baik digunakan pada aplikasi FTP jika koneksi antar router yang membentuk *tunnel* tidak stabil dan sering mengalami gangguan interferensi yang menyebabkan paket hilang saat pengiriman file.

Kata kunci : IPv6, Tunneling, GRE, ISATAP, *throughput*, *delay, packet loss,* FTP.

ABSTRACT

Name : Ghiffari Aulia
Study Program : Computer Engineering
Title : Performance Comparison on IPv6 Tunneling Network between GRE and ISATAP on FTP

This thesis will design a testbed to measure file transfer performance on IPv6 tunneling GRE and ISATAP including the comparison. IPv6 tunneling is a feature in IPv6 to help network migrate from IPv4 network to IPv6 network gradually. In the process of integration, choosing the type of tunneling has to be based on running application on the network. This paper aims to compare two type of tunneling (GRE and ISATAP) on FTP by referring to their QoS parameters. These QoS parameters are important to select the best tunneling type between GRE and ISATAP when transporting TCP packets.

ISATAP tunneling throughput on the network has increased by 0.15% of the throughput on the network GRE tunneling. Delay is also not much different, the ISATAP tunneling delay only decreased by 0.98%. Significant differences occurred in which the ISATAP packet loss have a greater packet loss, which is 7.488% as compared to the GRE packet loss is 5.562%. Therefore, GRE tunneling is better used on FTP when connection between the routers making the tunnel is not stable and often get interference that can make packet lost during file transfer.

Key words: IPv6, Tunneling, GRE, ISATAP, throughput, delay, packet loss, FTP.

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1. Latar Belakang

Perkembangan jaringan komputer yang sangat cepat membuat aplikasi dan konten-konten yang berada di internet semakin beragam. Dengan adanya hal itu, kepopuleran jaringan IP (*Internet Protocol*) semakin meningkat dibandingkan jaringan lain, seperti AppleTalk, IPX, dll. Saat ini, pertumbuhan pengguna internet semakin meningkat. Sejalan dengan hal itu, pemakaian alamat IP juga meningkat tajam hingga alamat IP yang tersisa tinggal sedikit dan diperkirakan tidak akan cukup memenuhi semua pengguna internet.

Untuk mengatasi masalah kekurangan alamat IP, *Internet Engineering Task Force* (IETF) mengembangkan protokol baru, yaitu *Internet Protocol version 6* (IPv6). Sebelumnya, IETF memberikan fitur *Network Address Translation* (NAT) untuk mentranslasikan alamat IP privat (RFC 1918) ke alamat IP publik untuk penghematan pemakaian alamat IP di internet. Namun, hal tersebut masih sangat jauh dari penyelesaian kekurangan IP dalam waktu yang lama. Untuk itu, IETF menyiapkan IPv6 sebagai solusi untuk 20 tahun mendatang.

IPv6 menggunakan jumlah bit yang lebih besar dibandingkan IPv4. Pada IPv4, total jumlah bit yang digunakan adalah 32 sedangkan untuk IPv6 adalah 128. Dengan begitu total alamat yang tersedia adalah 2^{128} alamat. Jumlah tersebut jauh lebih besar dari jumlah alamat IPv4 yang hanya berjumlah 4.294.967.296 alamat. Oleh karena itu, IPv6 tidak membutuhkan alamat privat sehingga semua divais mempunyai alamat publik dan pengiriman data tidak memiliki *latency* akibat proses translasi oleh NAT.

Selain itu, IPv6 juga memiliki *header* yang lebih sedikit daripada IPv4. IPv6 hanya mempunyai 7 *field* sedangkan IPv4 mempunyai 13 field. *Header* IPv6 juga memiliki fitur-fitur lebih dibanding IPv4, salah satunya adalah fitur keamanan. Berbeda dengan IPv4 yang masih menggunakan protokol yang terpisah (IPSec).

Dalam kenyataannya, proses migrasi dari jaringan IPv4 ke jaringan IPv6 tidaklah mudah. Hal ini disebabkan oleh banyaknya divais yang memakai alamat IPv4. Tidak mungkin seluruh divais di dunia diganti konfigurasinya secara

bersamaan. Walaupun memungkinkan, akan besar dampaknya pada proses bisnis di dunia yang pada akhirnya akan membuat kekacauan karena proses bisnis berhenti secara global. Untuk itu, IETF tidak begitu saja mengembangkan IPv6 tanpa ada fitur agar IPv6 dapat dipakai bersamaan dengan IPv4 sehingga migrasi jaringan IPv6 ke jaringan IPv4 dapat dilakukan secara berangsur-angsur.

Fitur yang diberikan adalah dual stack, yaitu memakai alamat IPv4 dan IPv6 secara bersamaan, *tunneling*, dan *Network Address Translation-Protocol Translation* (NAT-PT). *Tunneling* adalah fitur yang menghubungkan jaringan-jaringan yang menggunakan IPv6 yang terpisah oleh jaringan yang menggunakan IPv4. Beberapa metode tunneling yang ada adalah:

- *Manually Configured Tunnel* (MCT)
- *Generic Routing Encapsulation* (GRE)
- 6to4 *tunneling*
- *Intra-Site Automatic Tunneling Addressing Protocol* (ISATAP)
- Teredo

1.2. Tujuan Penulisan

Penulisan skripsi ini bertujuan untuk menganalisis performansi aplikasi transfer file pada jaringan yang menggunakan tunneling GRE dan ISATAP menggunakan *throughput*, *delay,* dan *packet loss* saat aplikasi dijalankan.

1.3. Perumusan Masalah

Pada Skripsi ini, akan dianalisis perbandingan performansi aplikasi FTP pada jaringan *tunneling* IPv6. Jaringan *tunneling* IPv6 yang dipakai untuk perbandingan adalah GRE dan ISATAP. Metode *tunneling* ini akan menghubungkan jaringan-jaringan yang memakai alamat IPv6 melewati jaringan yang menggunakan alamat IPv4. Masalah yang akan dibahas pada skripsi ini adalah:

1. Cara pengalamatan IPv6
2. Proses *tunneling* GRE dan ISATAP pada jaringan IPv6
3. Perbandingan performansi aplikasi transfer file pada *tunneling* GRE dan ISATAP

1.4. Batasan Masalah

Pada skripsi ini, penelitian dilakukan mulai dari tahap implementasi jaringan IPv4 murni, jaringan IPv6 murni, jaringan *tunneling* IPv6 GRE, jaringan *tunneling* IPv6 ISATAP, beserta *server* dan *client* untuk aplikasi transfer file. Pada skripsi ini, akan dianalisis perbandingan performansi FTP pada implementasi jaringan yang disebutkan sebelumnya.

1.5. Sistematika Penulisan

Penulisan Tugas Akhir ini terdiri dari 6 bab dengan sistematika penulisan sebagai berikut:

Bab I adalah pendahuluan. Pada bab ini akan dibahas tentang latar belakang, tujuan, perumusan masalah, pembatasan masalah, dan sistematika penulisan.

Bab II akan menjelaskan tentang landasan-landasan teori mengenai konsep IPv6 yang terdiri dari macam-macam alamat pada IPv6, *subnetting*, pengalamatan, format *header* pada paket IPv6, sistem keamanan IPv6, *tunneling* IPv6, dan aplikasi transfer file.

Selanjutnya adalah Bab III dimana akan dibahas konfigurasi dan implementasi jaringan yang akan diambil data-datanya.

Pada Bab IV, dari skripsi ini akan memberikan data-data yang diambil dari aplikasi FTP yang dijalankan pada konfigurasi jaringan yang diatur. Setelah itu, analisis akan dilakukan untuk *throughput*, *delay*, dan *packet loss* pada jaringan IPv6 tunneling GRE, dan jaringan IPv6 ISATAP.

Setelah penjabaran analisis, kesimpulan akan diberikan pada Bab V tentang hasil dari penelitian yang dilakukan berdasarkan data-data yang didapat dari jaringan yang dibangun.

Bagian terakhir akan melampirkan konfigurasi-konfigurasi dari divais-divais yang digunakan dan daftar acuan beserta daftar pustaka yang dirujuk.

BAB II

IPv6, IPv6 *Tunneling*, dan FTP

2.1. IPv6

2.1.1. Agregasi Rute Jaringan Global

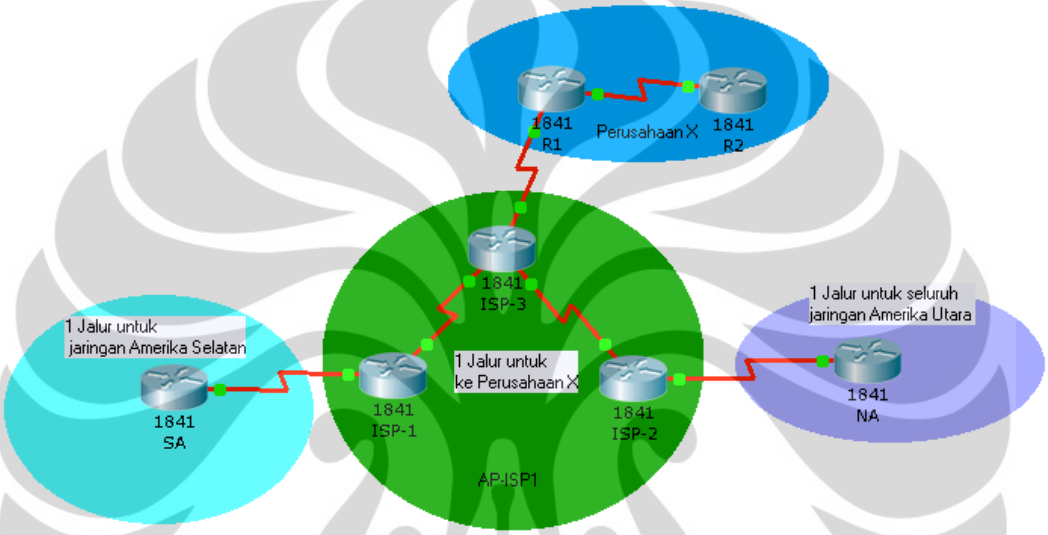
Pada saat komunitas Internet mulai serius untuk menemukan solusi untuk masalah pertumbuhan Internet, banyak pakar telah sepakat bahwa kebijakan alamat internet global IPv6 bisa membantu menjaga Internet tabel *routing* lebih kecil dan lebih mudah dikelola. Tugas alamat IPv6 mengikuti konsep tersebut.

Strategi Pengalamatan IPv6 dapat dijabarkan sebagai berikut:

- alamat IPv6 Publik dikelompokkan (numerik) menurut wilayah geografis.
- Di dalam masing-masing daerah, ruang alamat dibagi oleh ISP dalam wilayah itu.
- Di dalam masing-masing ISP di suatu daerah, ruang alamat lebih dibagi untuk setiap pelanggan.

Organisasi-organisasi yang sama menangani alamat IPv6 sebagaimana untuk IPv4. The Internet Corporation for Assigned Network Numbers (ICANN, www.icann.org) memiliki proses, dengan Internet Assigned Numbers Authority (IANA) mengelola proses. IANA memberikan satu atau lebih rentang alamat IPv6 untuk setiap *Regional Internet Registries* (RIR), yang berjumlah 5, meliputi Amerika Utara, Amerika Tengah / Selatan, Eropa, Asia / Pasifik, dan Afrika. RIR tersebut kemudian membagi alamat yang diberikan kepada pengguna menjadi bagian yang lebih kecil, menetapkan prefiks ke ISP yang berbeda dan pendaftar yang lebih kecil lainnya, dengan ISP kemudian menetapkan rentang lebih kecil dari alamat mereka kepada pelanggan ISP tersebut.

Rencana Pengalamatan IPv6 global dalam *routing table* lebih efisien, seperti contoh yang ditunjukkan pada Gambar 2.1. Angka ini menunjukkan perusahaan fiktif (Perusahaan X) yang telah ditetapkan prefix IPv6 oleh ISP, AP-ISP1 (ISP-1 untuk Regional Asia Pasifik). Gambar menunjukkan jaringan Asia-Pacific Network Information Centre (APNIC), yang merupakan RIR untuk wilayah Asia Pasifik.



Gambar 2.1 Agregasi jalur pada IPv6

Seperti terlihat dalam Gambar 2.1, *router* yang dipasang oleh ISP di wilayah lain di dunia dapat memiliki satu rute yang cocok dengan semua alamat IPv6 di Asia Pasifik. Meskipun mungkin ada ratusan ISP yang beroperasi di Asia Pasifik dan ratusan ribu ISP yang lebih kecil yang berada di bawah ISP tersebut, dan puluhan juta pengguna internet perorangan, semua alamat IPv6 publik dapat dari satu (atau beberapa) alamat blok yang sangat besar hanya membutuhkan satu (atau beberapa) rute pada router internet di bagian lain dunia. Demikian pula, dalam router ISP lainnya di Asia Pasifik (misalnya, AP-ISP2, ISP nomor 2 dalam Gambar 2.1), dapat memiliki satu rute yang cocok dengan semua rentang alamat untuk AP-ISP1. Router dalam AP-ISP1 hanya perlu memiliki satu rute yang cocok dengan rentang alamat yang diberikan kepada Perusahaan X daripada perlu mengetahui tentang semua *subnet* dalam Perusahaan X.

Selain menjaga tabel *routing* yang jauh lebih kecil, proses ini juga menghasilkan perubahan yang lebih sedikit untuk tabel *routing* Internet. Misalnya, jika AP-ISP1 menandatangani kontrak layanan dengan pelanggan lain, AP-ISP1 bisa menetapkan prefiks lain dalam berbagai alamat yang sudah diberikan untuk AP-ISP1 oleh APNIC. Router di luar AP-ISP1 tidak perlu tahu rute baru, karena rute yang ada sudah sesuai dengan rentang alamat yang diberikan ke pelanggan baru. *Router* ISP2 memiliki rute yang sesuai dengan rentang alamat keseluruhan dengan AP-ISP1, sehingga AP-ISP2 tidak perlu rute lagi ke pelanggan baru . Demikian juga, router di ISP di Eropa dan Amerika Utara sudah memiliki rute ke dalam semua *subnet* yang berada dalam rentang alamat AP-ISP1.

2.1.2. Format Alamat

Konvensi IPv6 menggunakan 32 angka heksadesimal, disusun dalam 8 kuartet dari 4 digit hexadesimal dipisahkan oleh titik dua untuk mewakili alamat 128-bit IPv6, misalnya:

2340:1111:AAAA: 0001:1234:5678:9ABC

Setiap digit hex mewakili 4 bit, jadi jika ingin memeriksa alamat dalam biner.

Untuk menyederhanakan pengalamatan, dua konvensi telah ditetapkan untuk mempersingkat penulisan untuk alamat IPv6:

- Menghilangkan 0 yang berada di dalam sebuah kuartet
- Mewakilkan satu atau lebih berturut-turut kuartet dari semua 0 hex dengan "::" tapi hanya untuk satu deret kuartet 0.

Sebagai contoh, perhatikan alamat berikut. Angka tebal mewakili digit di mana alamat bisa disingkat.

FF00: 0000:0000:0001:0000:0000:0000:0020

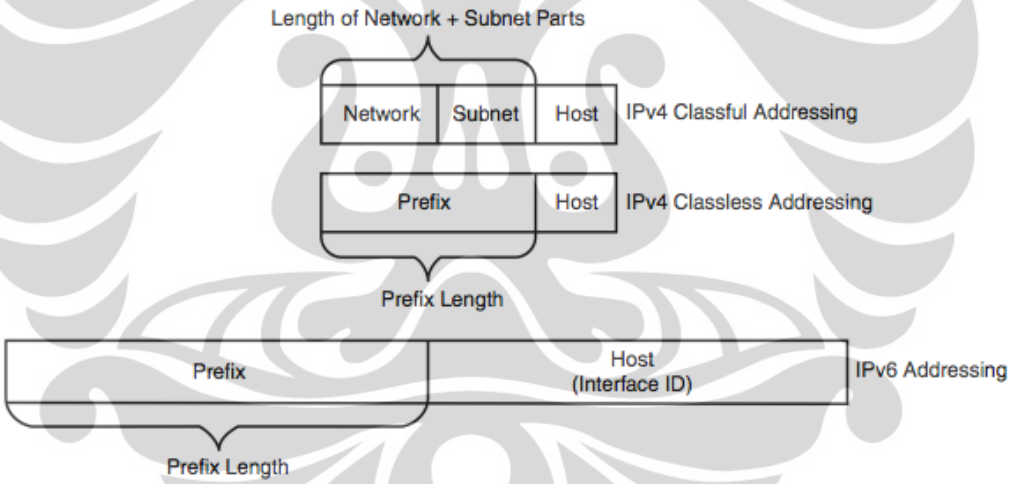
Alamat ini memiliki dua lokasi yang berbeda di mana satu atau lebih kuartet memiliki empat angka hexadesimal 0, jadi dua pilihan utama ada untuk menyingkat alamat menggunakan singkatan :: dalam satu atau lokasi lainnya. Dua opsi berikut menunjukkan dua singkatan yang diperbolehkan:

- FF00 ::1:0:0:0:20
- FF00:0:0:1:: 20

Secara khusus tanda "::" yang berarti satu atau lebih kuartet yang semua digitnya bernilai 0 tidak dapat digunakan dua kali karena itu akan menjadi ambigu bagi router jika menerima paket dengan alamat tujuan tersebut. Jadi, singkatan FF00::1::20 tidak dapat digunakan.

2.1.3. Konvensi untuk Menulis Prefiks IPv6

Prefiks IPv6 mewakili berbagai atau suatu blok alamat IPv6. Sama seperti router menggunakan IPv4 subnet IPv4 di tabel *routing* untuk mewakili rentang alamat tertentu, router menggunakan prefiks IPv6 untuk mewakili rentang alamat IPv6 juga. Konsep ini sama dengan pengalamatan *classless* pada alamat IPv4. Gambar 2.2 menggambarkan konsep pengalamatan IPv4 dan IPv6.



Gambar 2.2 pengalamatan *classful* dan *classless* IPv4 dan pengalamatan IPv6 [1]

Pertama, untuk perspektif, membandingkan pandangan classful dan tanpa kelas alamat IPv4. Pengalamatan *classful* IPv4 berarti bahwa sebuah jaringan mempunyai identitas berdasarkan kelas dan merupakan bagian dari kelas yang lebih besar. Sebagai contoh, alamat 128.107.3.0/24 (atau 128.107.3.0 255.255.255.0) berarti porsi bit jaringan 16 (karena alamat tersebut dalam jaringan kelas B), 8 bit *host* (karena memiliki 8 biner 0 pada *subnet mask*), dan menyisakan 8 bit untuk *subnet*. Nilai yang sama,

ditafsirkan dengan aturan *classless*, berarti prefiks 128.107.3.0 mempunyai panjang prefiks 24.

IPv6 menggunakan pengalamatan *classless*. Seperti IPv4, prefiks IPv6 berupa alamat jaringan, garis miring, dan kemudian panjang prefiks. Seperti IPv4 prefiks, bagian terakhir dari bit untuk *host*, bit awalan, akan diwakili oleh biner 0. Dan akhirnya, alamat prefiks IPv6 dapat disingkat dengan aturan yang sama seperti alamat IPv4.

Sebagai contoh, perhatikan alamat IPv6 berikut yang diberikan ke *host* pada sebuah LAN: 2000:1234:5678:9ABC:DE1:2345:6789:ABCD/64

Nilai ini mewakili 128-bit dari alamat IPv6 dan alamat ini tidak bisa disingkat. Namun, tanda /64 berarti bahwa *subnet* di mana alamat ini berada dalam satu jaringan yang mempunyai 64 bit pertama yang sama dengan alamat *host* lain dalam jaringan yang sama. Secara konseptual, itu adalah logika yang sama seperti alamat IPv4, misalnya, 128.107.3.1/24 alamat di awalan (subnet) yang pertama 24 bit adalah nilai-nilai yang sama seperti alamat 128.107.3.1.

Seperti dengan IPv4, ketika menulis sebuah prefix, bit yang berada di luar panjang prefix diberikan nilai bit 0. Dalam alamat IPv6 sebelumnya menunjukkan, awalan di mana alamat berada adalah

2000:1234:5678:9ABC: 0000:0000:0000:0000/64

dan ketika disingkat, akan menjadi:

2000:1234:5678:9ABC::/64

Jika panjang prefiks bukan kelipatan dari 16, maka batas antara *prefix* dan *host ID* berada di dalam sebuah kuartet. Dalam kasus tersebut, nilai prefiks harus daftar semua nilai dalam oktet terakhir di bagian awalan dari nilai. Misalnya, jika alamat memiliki panjang *prefix* / 56, bit jaringan akan mencakup semua 3 kuartet pertama (total 48 bit), ditambah 8 bit pertama dari kuartet keempat. The 8 bit berikutnya (terakhir 2 digit hex) dari oktet keempat diisi oleh biner 0, sebagai bagian dari bagian host dari alamat. Jadi, sisa oktet keempat harus ditulis, setelah diatur ke 0 biner, sebagai berikut:

2000:1234:5678:9A00 :: / 56

Daftar berikut merangkum beberapa poin penting tentang bagaimana menulis prefiks IPv6.

- *Prefix* memiliki nilai yang sama dengan alamat IP dalam rentang alamat dalam panjang *prefix*.
- Setiap bit setelah jumlah panjang *prefix* bit diisi biner 0.
- *Prefix* dapat disingkat dengan aturan yang sama seperti alamat IPv6.
- Jika panjang prefix tidak pada batas kuartet, tuliskan kuartet tersebut seluruhnya.

Bit-bit 0 tidak dapat jika berada di bagian belakang dalam kuartet . Sebagai contoh, 5000::/3 tidak diperbolehkan disingkat menjadi 5::/3, karena menghilangkan sisa oktet, dan perangkat jaringan tidak akan tahu apakah 5::/3 berarti "hex 0005" atau "hex 5000".

2.1.4. Subnetting

Konsep *subnetting* pada IPv6 sama dengan yang ada di IPv4, yaitu membagi alamat jaringan yang diberikan menjadi jaringan-jaringan yang lebih kecil sesuai perspektif berikut:

- *Prefix* yang diberikan kepada suatu pelanggan berlaku seperti *prefix* yang diberikan pada IPv4.
- Dalam melakukan *subnetting*, bit pada *host* dipinjam untuk membuat *subnet.*
- Bit yang tersisa digunakan sebagai ID untuk *host*.

Sebagai contoh, diberikan alamat jaringan 2001:0:0::/48 kepada sebuah perusahaan. Dalam merancang jaringan, alamat yang diberikan akan dibagi menjadi beberapa *subnet* yang diperlukan untuk setiap VLAN, *link* serial, dan jaringan WAN. Misalnya jaringan yang akan dibangun mempunyai topologi sebagai berikut.

48 bit	16 bit	64 bit
Prefix (diberikan oleh ISP) 2001:0:0	Subnet	*Host*

Gambar 2.3 Format *prefix*

Gambar 2.3 di atas menunjukkan alamat yang diberikan oleh ISP (/48) diperpanjang menjadi /64 sehingga terdapat 16 bit yang digunakan sebagai porsi *subnet* dan 64 bit untuk porsi *host*. Konsep ini sama dengan yang ada di IPv4, dimana untuk membuat *subnet*, bit pada *host* dipinjam.

Jumlah *subnet* yang dapat dibuat dengan meminjam 16 bit dari *host* adalah sebanyak 2^{16} dan jumlah *host* tiap *subnet* adalah sebanyak 2^{64}. Terlihat dari porsi yang diberikan tampaknya sangat besar dan tidak efisien. Namun, porsi yang sangat besar tersebut dapat dimanfaatkan untuk mengaktifkan fitur pengalamatan otomatis pada alamat IPv6 global *unicast*.

Setelah alamat IPv6 yang diberikan telah dibagi, alamat *subnet* dapat diberikan kepada LAN maupun WAN yang ada. Alamat *subnet* yang dapat dipakai sesuai contoh di atas berada pada rentang 2001::/64 sampai 2001:0:0:FFFF::/64.

2.1.5. Terminologi *Prefix*

Prefix atau alamat IPv6 yang diberikan mempunyai beberapa istilah sesuai penggunaannya. Tabel 2.1 merangkum istilah yang dimaksud.

Tabel 2.1 Terminologi *prefix*

Istilah	Diberikan oleh
Registry prefix	IANA ke *Regional Internet Registries* (RIR)
ISP *prefix*	RIR ke ISP
Site prefix	ISP ke pelanggan
Subnet prefix	Pelanggan ke jaringan internal

2.1.6. Jenis-Jenis Alamat IPv6

Selain alamat global *unicast*, IPv6 mempunyai beberapa alamat tipe lain sebagai berikut:

■ *Unicast*

Seperti pada IPv4, alamat *unicast* digunakan untuk mengidentifikasi *host* atau antarmuka dari router dalam komunikasi data. Alamat *unicast* terdiri dari alamat global, alamat lokal, alamat *link local*, alamat *site local*, *unspecified local*, dan alamat *loopback*.

■ *Multicast*

Alamat *multicast* digunakan untuk mewakili kelompok-kelompok *host* sehingga sebuah *host* dapat mengirimkan paket yang dikirimkan untuk semua *host* yang berada dalam kelompok alamat *multicast*. Alamat *multicast* mempunyai format seperti Gambar 2.4.

Gambar 2.4 Format alamat *multicast*[2]

Alamat *multicast* mempunyai 8 bit pertama berupa “11111111” yang ditunjukkan dengan “FF” pada 2 angka hex pertama di alamat IPv6. Alamat ini tidak dapat disebarkan dalam proses *routing*. Tabel 2.2 menunjukkan alamat-alamat *multicast* yang sering dipakai:

Tabel 2.2 Alamat *multicast*[12]

Penggunaan	Alamat IPv6	Alamat IPv4 yang ekivalen
Semua *node* pada jaringan	FF02::1	Alamat *broadcast* pada jaringan
Semua *router* pada jaringan	FF02::2	-
Paket *routing* OSPF	FF02::5, FF02::6	224.0.0.5, 224.0.0.6
Paket *routing* RIP-2	FF02::9	224.0.0.9
Paket *routing* EIGRP (protokol cisco)	FF02::A	224.0.0.10
DHCP *relay agent*	FF02::1:2	-
Server DHCP	FF05::1:3	-
Semua server NTP	FF05::101	-

- *Anycast*

  Alamat *anycast* pergunakan pada banyak antarmuka dan dapat mengidentifikasi banyak alamat sehingga bisa digunakan untuk menduplikasi sebuah server.

2.1.7. Metode Pengonfigurasian Alamat IPv6

Dalam mengonfigurasi alamat IPv6, dibutuhkan informasi alamat, *subnet mask*, *default gateway*, dan alamat IPv6 dari server DNS. Ada beberapa cara yang digunakan yang digunakan untuk melakukan konfigurasi dimulai dari otomatis maupun manual[12].

- *Stateful* DHCP

  Pengalamatan dengan metode ini berupa pengalamatan secara otomatis. Untuk itu diperlukan sebuah *stateful* server DHCP yang memberikan semua informasi yang dibutuhkan oleh *host*.

- *Stateless autoconfig*

  Pada metode ini, *host* akan mengirimkan paket *Router Solicitation* (RS) ke router untuk mendapatkan alamat IPv6 router. Selanjutnya, router akan membalas dengan paket *Router Advertisement* (RA) yang berisi informasi alamat IPv6 dari router. Setelah mendapatkan alamat router, *host* akan mendapatkan alamat IPv6 dengan mengisi ID antarmuka dengan alamat MAC dari kartu jaringan.

- *Static configuration*

  Konfigurasi ini sama dengan yang ada pada alamat IPv4, yaitu dengan mengonfigurasi secara manual alamat IPv6 pada *host*.

- *Static configuration* dengan EUI-64

  Metode ini hampir sama dengan konfigurasi statik namun yang perlu dikonfigurasi adalah alamat *prefix* saja. Sementara untuk alamat *host* akan didapatkan dengan metode EUI-64 yang mengambil ID dari alamat MAC.

2.1.8. Protokol *Routing* IPv6

Untuk mengurusi masalah *routing* pada IPv6, organisasi internet seperti IETF dan beberapa vendor mengeluarkan protokol versi baru dari protokol IPv4 yang sekarang. Protokol-protokol tersebut adalah:

- RIPng

  RIPng untuk IPv6 berbasis pada RIPv2, tetapi bukan merupakan ekstensi dari RIPv2 RIPng merupakan protokol terpisah[3]. RIPng tidak mendukung IPv4, jadi untuk menggunakan RIP untuk proses *routing* IPv4 dan IPv6 harus menggunakan RIPv1 atau RIPv2 untuk IPv4 dan RIPng untuk IPv6.

  RIPng menggunakan *timer*, prosedur, dan tipe pesan yang sama dengan RIPv2. Misalnya, RIPv2 menggunakan *update timer* 30 detik yang telah ditambahi sedikit untuk mencegah sinkronisasi, periode *timeout* 180 detik dan holddown timer 180 detik. RIPng juga menggunakan metrik menghitung hop sama seperti

protokol RIP pada IPv4, dengan 16 menunjukkan jalur yang tidak dapat dijangkau dan juga menggunakan pesan *request* dan *response* dengan cara yang sama seperti RIPv2. Serta pesan *request* dan *response* dikirim secara *multicast* dengan sedikit pengecualian untuk *unicast* yang digunakan RIPv1 dan RIPv2. Address multicast IPv6 yang digunakan RIPng adalah FF02::9.

- OSPFv3

  OSPFv3 ditetapkan pada RFC 2740[4]. OSPFv3 menggunakan mekanisme fundamental yang sama dengan yang ada di IPv4, yaitu OSPFv2. OSPFv3 juga menggunakan algoritma *Shortest Path First* (SPF), *flooding,* dan pemilihan *Designated Router* (DR)/*Backup Designated Router* (BDR) pada jaringan *multi-access*.

- EIGRP

  EIGRP merupakan protokol yang dikembangkan oleh Cisco Systems, Inc pada jaringan IPv6. Oleh karena itu, protokol ini tidak dapat digunakan selain di jaringan yang memakai perangkat cisco.

- *Static Routing*

  Static routing adalah metode *routing* yang sederhana dan dilakukan manual oleh setiap router. Metode ini dipakai di skripsi untuk menjalurkan paket-paket dari LAN server ke LAN klien dan juga sebaliknya.

2.2. IPv6 *Tunneling*

Setiap institusi atau organisasi yang menggunakan IPv4 tidak akan memindahkan begitu saja jaringan internal mereka ke jaringan IPv6 secara masif dan total dalam sekali migrasi. Pada kenyataannya, setiap institusi atau organisasi

akan memindahkan jaringannya ke IPv6 satu per satu dimulai dari *subnet* yang membutuhkan jaringan IPv6 yang konsisten.

Butuh waktu yang lama untuk memindahkan semua jaringan sebuah institusi dari IPv4 ke IPv6. Dalam proses yang panjang tersebut, ada beberapa cara yang disediakan IPv6 dalam proses migrasi tersebut, yaitu:

- *Dual stacks*
- *Tunneling*
- NAT *Protocol Translator* (NAT-PT)

Dari ketiga fitur tersebut, *tunneling* merupakan salah satu fitur yang diminati dimana *tunnel* yang melewati jaringan IPv4 menghubungkan beberapa LAN yang menggunakan IPv6. Skripsi ini membahas tentang performansi kedua tipe *tunneling* ketika menggunakan aplikasi transfer file menggunakan *File Transfer Protocol* (FTP).

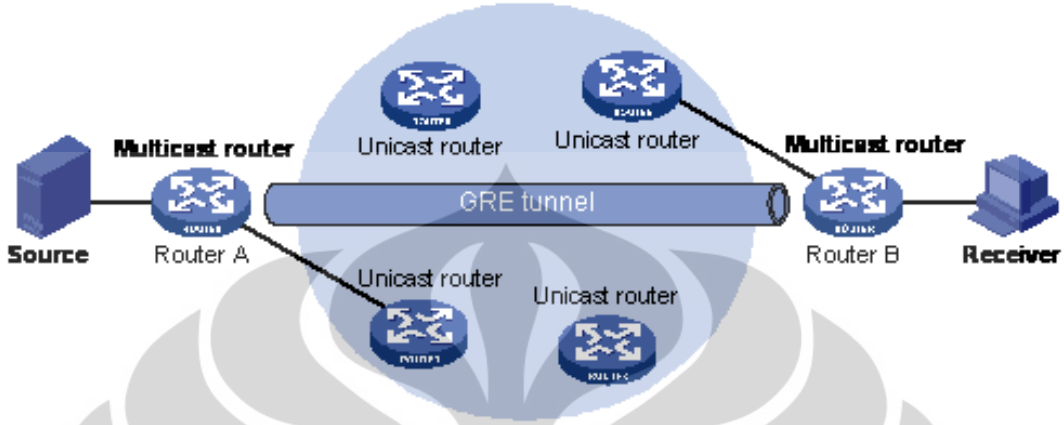
2.2.1. Generic Routing Encapsulation (GRE)

Generic Routing Encapsulation (GRE) adalah protokol enkapsulasi IP yang digunakan untuk membawa paket dalam jaringan. Paket-paket dikirim melalui sebuah *tunnel*.

GRE mengenkapsulasi paket ke paket IP dan mengarahkan paket tersebut ke sebuah *host* perantara, di mana akan didekapsulasi dan diteruskan ke tujuan akhir mereka. *Tunnel* GRE memungkinkan routing protokol seperti RIP dan OSPF untuk meneruskan paket data dari satu switch ke switch lain di Internet. Selain itu, *tunnel* GRE dapat mengenkapsulasi aliran *multicast* data untuk transmisi melalui Internet.

GRE dijelaskan dalam RFC 2784[5]. Sebagai router *tunnel source*, saklar merangkum paket payload untuk transportasi melalui *tunnel* ke jaringan tujuan. Paket *payload* pertama kali dikemas dalam sebuah paket GRE, dan paket GRE yang dihasilkan dikemas dalam sebuah protokol pengiriman. Switch yang melakukan fungsi *tunnel* mengekstrak paket *tunnel* dan meneruskan paket ke jaringan tujuan.

Secara spesifik, GRE dapat digunakan untuk *tunneling* IPv6. Dalam hal ini, router meminjam alamat IPv4 untuk membawa paket-paket dengan alamat IPv6.



Gambar 2.5 GRE *Tunneling* [6]

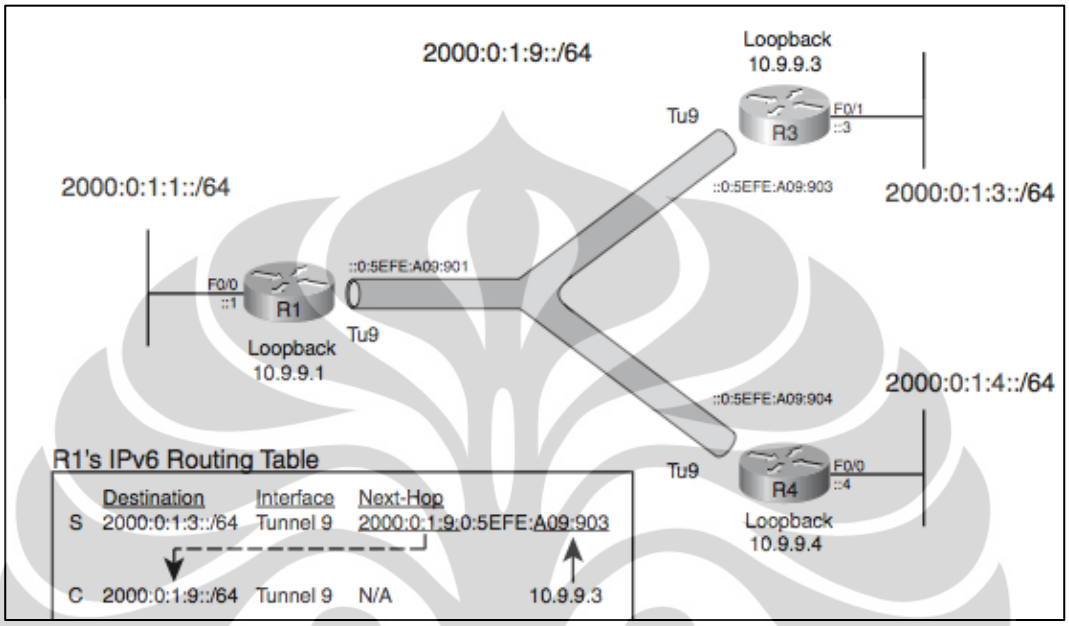
Gambar 2.5 menunjukkan *tunneling* pada GRE. Pada tunnel, walaupun paket melewati beberapa router yang memakai alamat IPv4 namun alamat yang dijadikan referensi adalah alamat IPv6. Dengan begitu, *tunnel* sumber dan *tunnel* tujuan mempunyai *prefix* yang sama.

GRE merupakan tipe *tunneling* yang bekerja secara *point-to-point*. Selain itu, GRE juga mendukung protokol-protokol *routing* seperti RIPng, OSPFv3, dan EIGRP.

2.2.2 *Intra-Site Automatic Tunneling Addressing Protocol* (ISATAP)

ISATAP merupakan protokol *tunneling* IPv6 yang dirilis tahun 2008. ISATAP dirancang untuk lingkup intra-situs. ISATAP menghubungkan *node* yang terisolasi dalam situs IPv4 melalui *tunnel* otomatis IPv6-IPv4. Itulah mengapa disebut intra-situs (*intra-site)* [7]. Dengan ISATAP, intra-situs jaringan IPv4 dipandang sebagai *link layer* untuk IPv6, dan *node* lain dalam jaringan intra-situs dianggap sebagai jaringan IPv6. Konsep *tunneling* otomatis juga didukung dimana mirip dengan model Non-Broadcast Multiple Access (NBMA). *Host* ISATAP mendapat 64-bit *prefix* dari server ISATAP. Kemudian alamat ISATAP dibentuk dengan id antarmuka sendiri (0:5EFE:alamat_IPv4). Setelah itu,

host ISATAP dapat terhubung satu sama lain melalui *tunnel* dengan alamat ISATAP. Selanjutnya, ISATAP dapat digunakan untuk menyediakan konektivitas ke jaringan IPv6 luar bersama-sama dengan mekanisme transisi lainnya.



Gambar 2.6 ISATAP *Tunneling*[8]

ISATAP merupakan tipe *tunneling* IPv6 yang bersifat *point-to-multipoint*. Seperti terlihat pada Gambar 2.6, untuk menjalurkan paket, router harus mengetahui IP *next-hop*. Untuk konfigurasi IPv6 pada *tunnel*, ISATAP mensyaratkan pengonfigurasian EUI-64 dengan format alamat seperti pada Gambar 2.6.

2.3. FTP

2.3.1. Protokol

File Transfer Protocol (FTP) adalah protokol jaringan standar yang digunakan untuk mentransfer file dari satu host ke host lain atau melalui jaringan berbasis TCP, seperti Internet.

FTP dibangun pada arsitektur klien-server dan menggunakan kontrol terpisah dan sambungan data antara klien dan server. FTP pengguna dapat mengotentikasi sendiri dengan menggunakan metode *plain-text*, biasanya dalam bentuk *username* dan *password*, tetapi dapat terhubung secara anonim jika server dikonfigurasi untuk kemungkinan itu.

Untuk transmisi aman yang menyembunyikan (enkripsi) username dan password, dan mengenkripsi konten, FTP sering menggunakan SSL / TLS (FTPS) dan SSH File Transfer Protocol (SFTP) sebagai gantinya.

Aplikasi FTP client pertama adalah baris perintah aplikasi yang dikembangkan sebelum sistem operasi memiliki antarmuka pengguna grafis, dan masih dikirimkan dengan Windows biasa, Unix, Linux dan sistem operasi lainnya.

2.3.2. Komunikasi Data FTP

Server merespon melalui koneksi kontrol dengan tiga digit kode status dalam ASCII dengan pesan teks opsional. Misalnya "200" (atau "200 OK") berarti bahwa perintah terakhir berhasil. Angka-angka mewakili kode untuk respon dan teks opsional merupakan penjelasan manusia-dibaca atau permintaan akun misalnya untuk menyimpan file. Sebuah perpindahan berkelanjutan file data melalui koneksi data dapat dibatalkan dengan menggunakan interrupt pesan yang dikirim melalui koneksi kontrol.

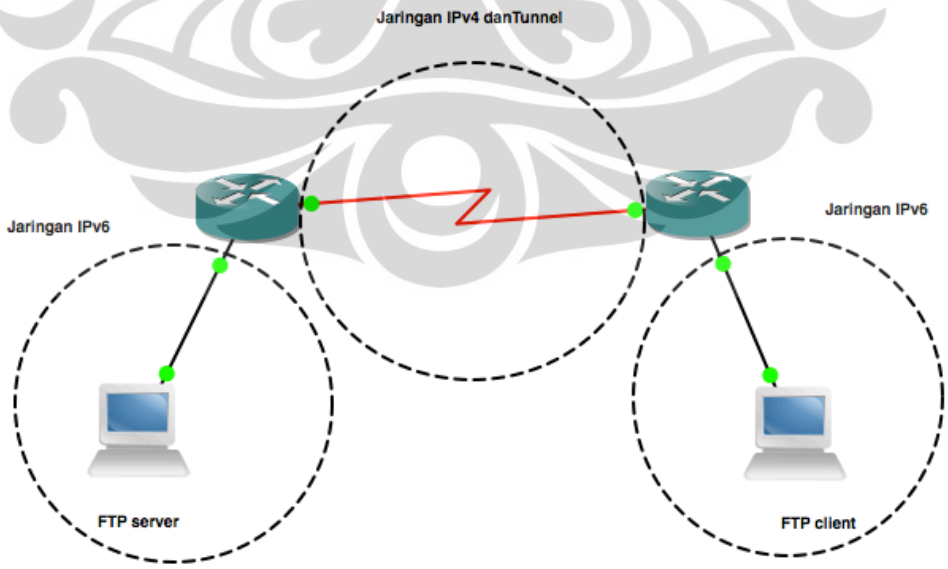
FTP dapat berjalan dalam mode aktif atau pasif, yang menentukan bagaimana koneksi data didirikan. Dalam modus aktif, klien membuat koneksi kontrol TCP ke server dan mengirimkan server alamat IP klien dan nomor *port* klien yang acak, dan kemudian menunggu sampai server memulai koneksi data melalui TCP ke alamat IP client dan nomor client port. Dalam situasi di mana klien berada di belakang firewall dan tidak dapat menerima koneksi masuk TCP, mode pasif dapat digunakan. Dalam modus ini, klien menggunakan koneksi kontrol untuk mengirim perintah PASV ke server dan kemudian menerima alamat IP server dan nomor port server dari server, dimana klien menggunakannya untuk membuka koneksi data dari sembarang *port* klien untuk alamat IP server dan nomor port server yang diterima.

BAB III
KONFIGURASI JARINGAN DENGAN APLIKASI TRANSFER FILE DAN METODE PENGAMBILAN DATA

3.1. Topologi Jaringan

Pada skripsi ini akan dibuat sebuah jaringan sederhana untuk menguji 2 *tunneling* IPv6. *Testbed* yang dibangun berupa jaringan sederhana yang menghubungkan dua LAN yang menggunakan alamat IPv6. Pada salah satu LAN terdapat 1 *Personal Computer* (PC) sebagai server FTP sedangkan pada LAN yang lain terdapat 1 *Personal Computer* (PC) sebagai client FTP. Setiap LAN terhubung oleh 1 router dan masing-masing router dihubungkan dengan kabel serial (T1). *Link* serial yang menghubungkan antar router menggunakan alamat IPv4. Di *link* inilah akan diimplementasi 2 tipe *tunneling* yang berbeda, yaitu *Generic Routing Encapsulation* (GRE) dan *Intra-Site Automatic Tunneling Addressing Protocol* (ISATAP).

Setiap tipe *tunneling* yang diimplementasi akan mempunyai jaringan dengan alamat IPv6. *Tunneling* dibentuk dengan mengonfigurasi alamat IPv6 pada antarmuka yang akan digunakan sebagai *tunnel* pada router. Gambar 3.1 menunjukkan topologi jaringan yang akan diimplementasikan.



Gambar 3.1 Topologi jaringan *testbed*

3.2. Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan pada topologi sesuai Gambar 3.1 yaitu:

3.2.1. Server FTP

Untuk server FTP, akan digunakan sebuah *Personal Computer* (PC) dengan spesifikasi sebagai berikut:

- Prosesor: Intel Core 2 Duo™
- Memori: 2 GB
- Harddisk: 250 GB

3.2.2. Client FTP

Untuk client FTP, akan digunakan sebuah *Personal Computer* (PC) dengan spesifikasi sebagai berikut:

- Prosesor: Intel Core 2 Duo™
- Memori: 2 GB
- Harddisk: 250 GB

3.2.3 Router LAN Server

Untuk router yang terhubung ke LAN server, akan digunakan sebuah router cisco seri 1841 seperti terlihat pada Gambar 3.2 dengan spesifikasi sebagai berikut:

- WIC, digunakan untuk menghubungkan kabel serial
- Memori: 256 MB SDRAM
- Flash: 64 MB

Gambar 3.2 Router cisco seri 1841

3.2.4. Router LAN Client

Untuk router yang terhubung ke LAN server, akan digunakan sebuah router cisco 1841 dengan spesifikasi sebagai berikut:

- WIC, digunakan untuk menghubungkan kabel serial
- Memori: 256 MB SDRAM
- Flash: 64 MB

3.2.5. *Link* Serial

Untuk menghubungkan antar 2 router digunakan sebuah kabel serial DTE (*Data Terminal Equipment*) dan DCE (*Data Circuit-Terminating Equipment*) dengan tipe V.35 dengan konektor *smart serial* seperti yang terlihat pada Gambar 3.3

Gambar 3.3 Kabel serial tipe V.35

3.3. Spesifikasi Perangkat Lunak

Secara umum, skripsi ini menggunakan beberapa perangkat lunak untuk mengimplementasi jaringan yang akan dibangun.

3.3.1. Sistem Operasi Linux 12.10 LTS

Seluruh PC yang digunakan untuk mengimplementasi jaringan *tunneling* ini menggunakan sistem operasi Linux Ubuntu 12.10 LTS. Sistem operasi ini digunakan karena bersifat *open source* sehingga dapat didapatkan secara gratis dan informasi mengenai aplikasi-aplikasi server sangat banyak dan dapat diatur mengikuti sistem yang dibuat untuk skripsi ini.

3.3.2. Cisco Interconnecting Operating System (IOS) 12.3

Cisco IOS digunakan karena perangkat keras yang digunakan adalah perangkat dari Cisco Systems, Inc. Untuk mengimplementasi jaringan IPv6 beserta *tunneling*-nya, cisco IOS minimal harus mempunyai versi 12.3. Tipe-tipe *tunneling* IPv6 yang didukung oleh IOS ini adalah GRE, 6to4, dan ISATAP.

3.3.3. ProFTPD

Program server FTP yang digunakan pada skripsi ini adalah ProFTPD. ProFTPD merupakan program server untuk aplikasi transfer file

yang bersifat *open source* dan banyak digunakan pada sistem operasi berbasis Unix dan Linux. Alasan digunakannya ProFTPD pada skripsi ini adalah karena ProFTPD merupakan program server FTP yang mendukung protokol IPv6.

3.3.4. FileZilla

FileZilla merupakan program client FTP *open source* yang berbasis *Graphical User Interface* (GUI). Tampilannya yang berupa GUI memudahkan pengguna untuk mengunduh atau mengunggah file ke server FTP tanpa harus paham perintah-perintah yang ada pada protokol FTP. FileZilla dapat digunakan di berbagai sistem operasi seperti, Unix, Linux, Windows, dan Macintosh. Selain digunakan untuk aplikasi FTP, FileZilla juga mendukung protokol-protokol transfer file lain seperti, *Secure File Transfer Protocol* (SFTP) dan *File Transfer Protocol over SSL* (FTPS).

3.3.5. Wireshark

Wireshark adalah *software* untuk analisis jaringan dan protokol komunikasi yang bersifat *free* dan *open source*. Dalam skripsi ini, wireshark digunakan untuk menangkap paket FTP yang keluar-masuk antarmuka server dan client. Wireshark memiliki fitur-fitur penting yang dibutuhkan untuk mengukur parameter-parameter QoS yang dibahas di skripsi ini. Wireshark mempunyai nama Ethereal sebelum berganti nama seperti sekarang dan tersedia untuk berbagai OS.

3.4. Konfigurasi Jaringan

3.4.1. Konfigurasi Alamat IPv6

Konfigurasi alamat IPv6 pada *testbed* dilakukan di PC server, PC client, dan kedua Router. Konfigurasi alamat pada router akan dibahas di subbab selanjutnya.

- Konfigurasi alamat IPv6 pada PC

  Untuk mengonfigurasi alamat IPv6 pada PC yang mempunyai sistem operasi Ubuntu 12.10 dapat dilakukan dengan 2 cara, yaitu dengan cara GUI dan mengedit file pada terminal. Untuk mengonfigurasi

menggunakan terminal, pertama kali kita harus membuka terminal di Ubuntu dan masuk sebagai *root* dengan perintah berikut.

```
# sudo -s
```

Kemudian lakukan tahap-tahap berikut:

1. Masuk direktori /etc/network
2. Ubah file interfaces
3. Atur alamat IPv6 secara static dan masukkan alamatnya beserta *netmask* dan alamat *gateway*.
4. *Restart* antarmuka jaringan

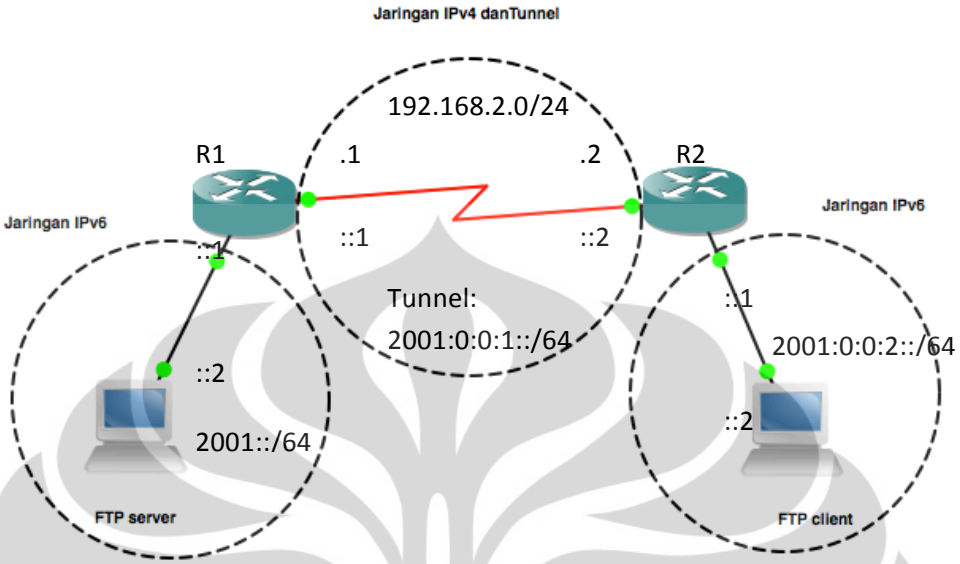
3.4.2. Konfigurasi Server FTP

Sebelum menggunakan ProFTPD, harus dilakukan instalasi dan mengunduh program tersebut dari repositori Ubuntu dengan perintah

```
# sudo apt-get install proftpd
```

Pengaturan pada ProFTD tidak akan diubah dan akan memakai konfigurasi awal yaitu menggunakan *default port* FTP (20/21).

3.4.3. Skenario *Tunneling* GRE



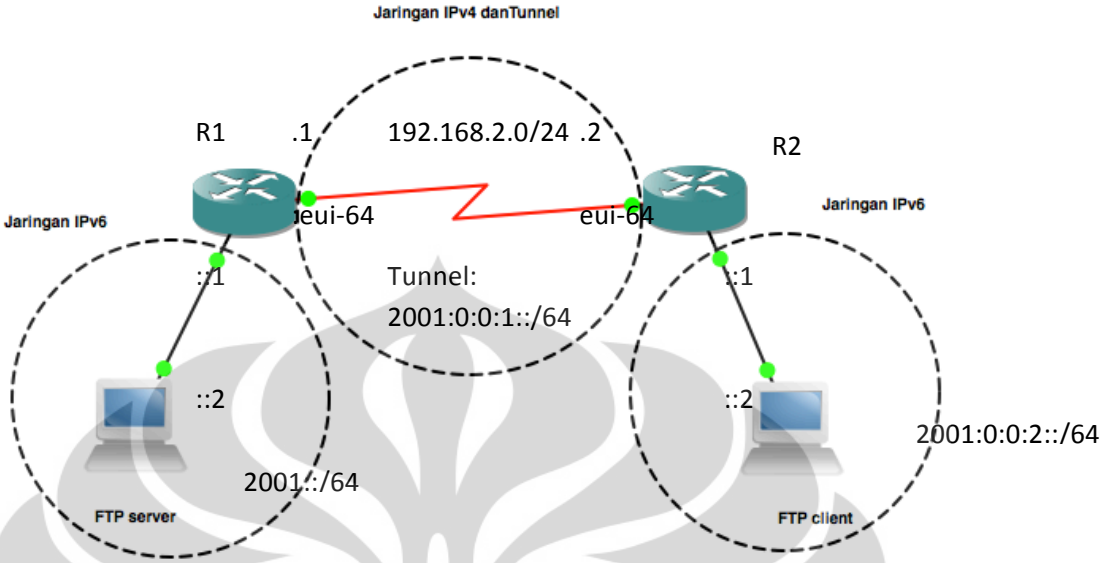
Gambar 3.4 Topologi Jaringan menggunakan *tunneling* GRE

Gambar 3.4 menunjukkan skenario pada jaringan yang menggunakan *tunneling* GRE. Terlihat bahwa pada LAN server dan LAN klien menggunakan jaringan IPv6 sedangkan router-router yang menghubungkan kedua LAN diberikan alamat IPv4 dan dikonfigurasi *tunnel*.

Paket dari LAN klien ke LAN server atau sebaliknya seakan-akan melewati jaringan 2001:0:0:1::/64. Sebenarnya yang dilewati adalah jaringan IPv4, yaitu 192.168.2.0/24. Setiap router dikonfigurasi antarmukanya dengan IPv4 atau IPv6 beserta antarmuka *tunnel*-nya. Antarmuka *tunnel* diberi alamat IPv6 sehingga ketika paket sampai pada router yang memiliki *tunnel*, paket akan dienkapsulasi ke dalam protokol GRE.

Contoh konfigurasi router dapat dilihat pada lampiran yang berada pada bagian akhir dari skripsi ini.

3.4.4. Skenario *Tunneling* ISATAP



Gambar 3.5 Topologi Jaringan menggunakan *tunneling* ISATAP

Pada skenario ini, alamat yang digunakan sama tetapi konfigurasi *tunneling*-nya menggunakan ISATAP. Terlihat bahwa pada LAN server dan LAN klien menggunakan jaringan IPv6 sedangkan router-router yang menghubungkan kedua LAN diberikan alamat IPv4 dan dikonfigurasi *tunnel*.

Paket dari LAN klien ke LAN server atau sebaliknya seakan-akan melewati jaringan 2001:0:0:1::/64. Sebenarnya yang dilewati adalah jaringan IPv4, yaitu 192.168.2.0/24. Setiap router dikonfigurasi antarmukanya dengan IPv4 atau IPv6 beserta antarmuka *tunnel*-nya. Antarmuka *tunnel* diberi alamat IPv6 sehingga ketika paket sampai pada router yang memiliki *tunnel*, paket akan dienkapsulasi ke dalam protokol ISATAP.

Berbeda dengan skenario yang menggunakan *tunneling* GRE, skenario ini menggunakan konfigurasi alamat IPv6 dengan metode EUI-64 untuk dapat membentuk *tunnel* seperti pada Gambar 3.5.

3.3. Metode Pengambilan Data

Setelah jaringan diimplementasi, akan dilakukan pengukuran parameter-parameter QoS pada setiap skenario. Pengukuran dilakukan pada saat mengunggah file dari klien ke server berukuran 1,1 MB dengan mengaktifkan wireshark agar dapat mengambil paket-paket FTP yang melewati *tunnel*. Wireshark diaktifkan untuk mendapatkan nilai *throughput* dan delay sedangkan untuk mendapatkan *packet loss*, wireshark juga harus diaktifkan pada sisi server. Percobaan ini sebelumnya juga dilakukan dengan mengirim satu file untuk mengukur performansi FTP pada tipe *tunneling* lain[9].

Pengambilan data dilakukan sebanyak 10 kali (10 sampel) agar didapatkan rata-rata parameter QoS yang akurat. Tidak ada perbedaan jika ingin mengambil data pada saat mengunduh atau saat mengunggah karena tidak ada perbedaan kondisi dalam hal ini tidak ada perbedaan *bandwidth*. Oleh karena itu, pengukuran dengan skenario mengunggah atau mengunduh hanya masalah arah saja.

Pengukuran ini difokuskan pada dua tipe *tunneling*, yaitu GRE dan ISATAP untuk mengetahui perbandingan performansi antara kedua tipe *tunneling* tersebut jika dipakai pada jaringan yang terdiri IPv4 dan IPv6. GRE merupakan *tunneling* universal yang tidak hanya dipakai untuk *tunneling* IPv6 saja sedangkan ISATAP adalah *tunneling* yang lebih baru dibandingkan GRE. Akan tetapi, belum ada yang membahas tentang perbandingan performansi GRE dengan protokol *tunneling* lain bahkan membandingkan dengan protokol yang lebih baru seperti ISATAP. FTP merupakan salah satu aplikasi yang banyak digunakan di Intenet. FTP dirancang untuk pengiriman file melalui TCP baik itu pada jaringan IPv4 maupun IPv6. Banyak file ditransfer menggunakan FTP dan jaringan masa depan akan memiliki konten yang lebih kaya lagi. Untuk itu, performansi FTP adalah krusial. Diperlukan analisis yang tepat untuk memilih implementasi jaringan. Analisis tersebut digunakan untuk mengidentifikasi faktor yang mempengaruhi performansi FTP[11]. Untuk jaringan IPv6 murni, performansi FTP lebih baik daripada menggunakan *tunneling*[9] sedangkan jaringan IPv4 murni tidak jauh berbeda dengan jaringan IPv6 murni seperti yang sudah pernah dilakukan[9].

BAB IV
ANALISIS PERFORMANSI FTP PADA JARINGAN *TUNNELING* IPv6

4.1. Pengujian FTP

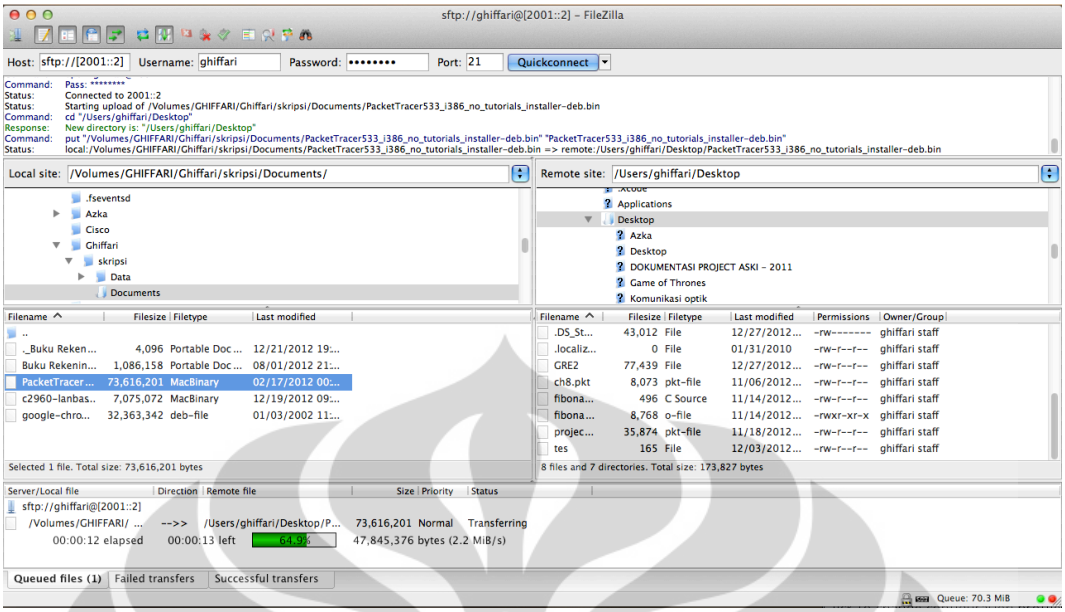
Pengujian performansi aplikasi transfer file dilakukan pada jaringan yang menggunakan *tunneling* untuk menghubungkan jaringan-jaringan IPv4. Pengalamatan dilakukan menggunakan alamat IPv6 global.

Pada pengujian kedua skenario yang menggunakan tipe *tunneling* yang berbeda, jaringan LAN klien dan jaringan LAN server tetap sama. Jaringan LAN server menggunakan alamat *prefix* IPv6 2001::/64 dan jaringan LAN klien mengugnakan alamat *prefix* IPv6 2001:0:0:2::/64.

Pada jaringan yang menghubungkan kedua LAN tersebut, alamat IPv4 dipergunakan dan *tunneling* akan diaktifkan. Untuk *tunneling* GRE, alamat global IPv6 yang diberikan adalah alamat IPv6 statik sedangkan untuk *tunneling* ISATAP menggunakan alamat IPv6 statik dengan metode EUI-64.

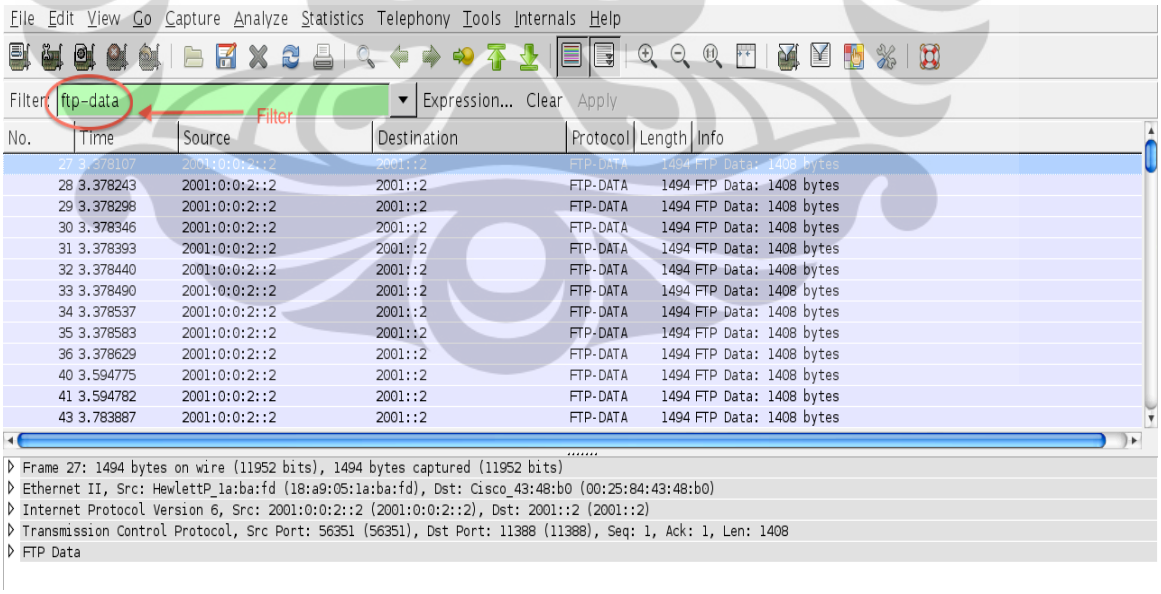
Sebelum dilakukan pengujian, komputer server dikonfigurasi menggunakan program ProFTPD sedangkan komputer klien diinstal program klien FTP yaitu FileZilla. Untuk mengukur performansi kedua aplikasi FTP pada kedua tipe *tunneling*, program wireshark diinstal pada komputer klien dan server. Wireshark diaktifkan sesaat sebelum file dikirim.

Pada saat pengujian, klien mengunggah sebuah file ke komputer server melewati *tunnel* yang sudah diimplementasi. Ukuran file yang dikirim adalah 1,1 MB untuk kedua skenario. Wireshark akan dinonaktifkan ketika file yang dikirim telah sepenuhnya sampai pada server dengan melihat FileZilla seperti pada Gambar 4.1.



Gambar 4.1 Mengunggah file menggunakan FileZilla

Untuk mendapatkan aspek kuantitas dari performansi, parameter-parameter QoS (*throughput*, *delay*, dan *packet loss*) akan diukur dari paket-paket yang ditangkap oleh wireshark. Proses pengiriman file dilakukan sebanyak 10 kali untuk mendapatkan hasil yang akurat dan data-data tersebut akan dirata-ratakan.



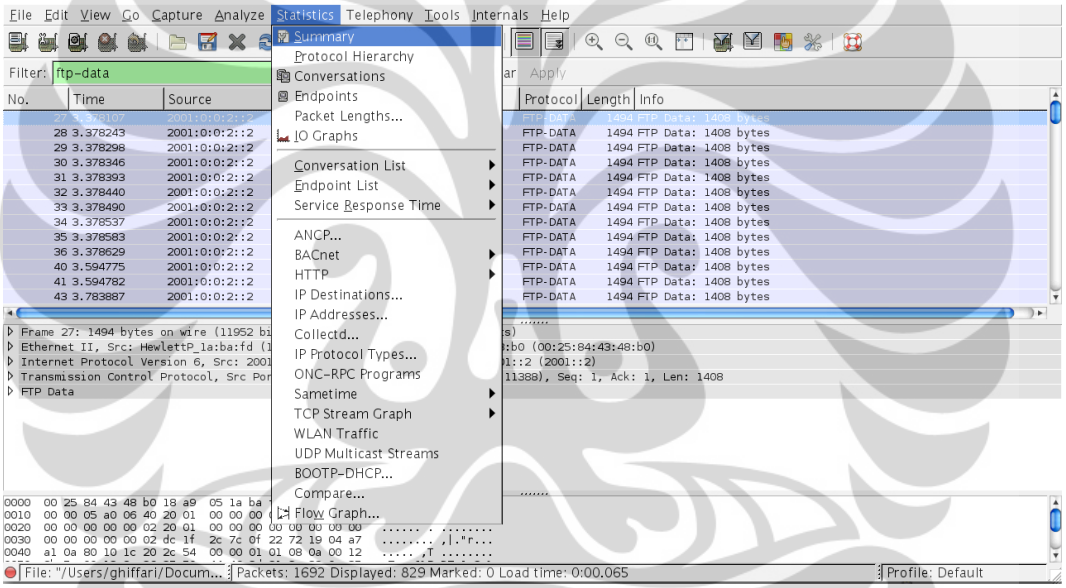
Gambar 4.2 Tampilan wireshark

Ketika file sudah selesai diunggah, wireshark akan menampilkan paket-paket yang ditangkap. Untuk menyaring paket FTP saja, wireshark harus ditambahkan ekspresi seperti pada Gambar 4.2

Selanjutnya, data mentah tersebut akan dilihat nilai dari parameter-parameter QoS, yaitu *throughput*, *delay,* dan *packet loss*.

4.2. Pengujian *Throughput*

Throughput adalah kecepatan transfer efektif pada sebuah kanal komunikasi data. *Throughput* diukur dengan satuan packet/sec atau bit/sec. Untuk pengukuran ini, satuan yang dipakai adalah kilobit/sec (Kbps).



Gambar 4.3 *Summary* pada wireshark

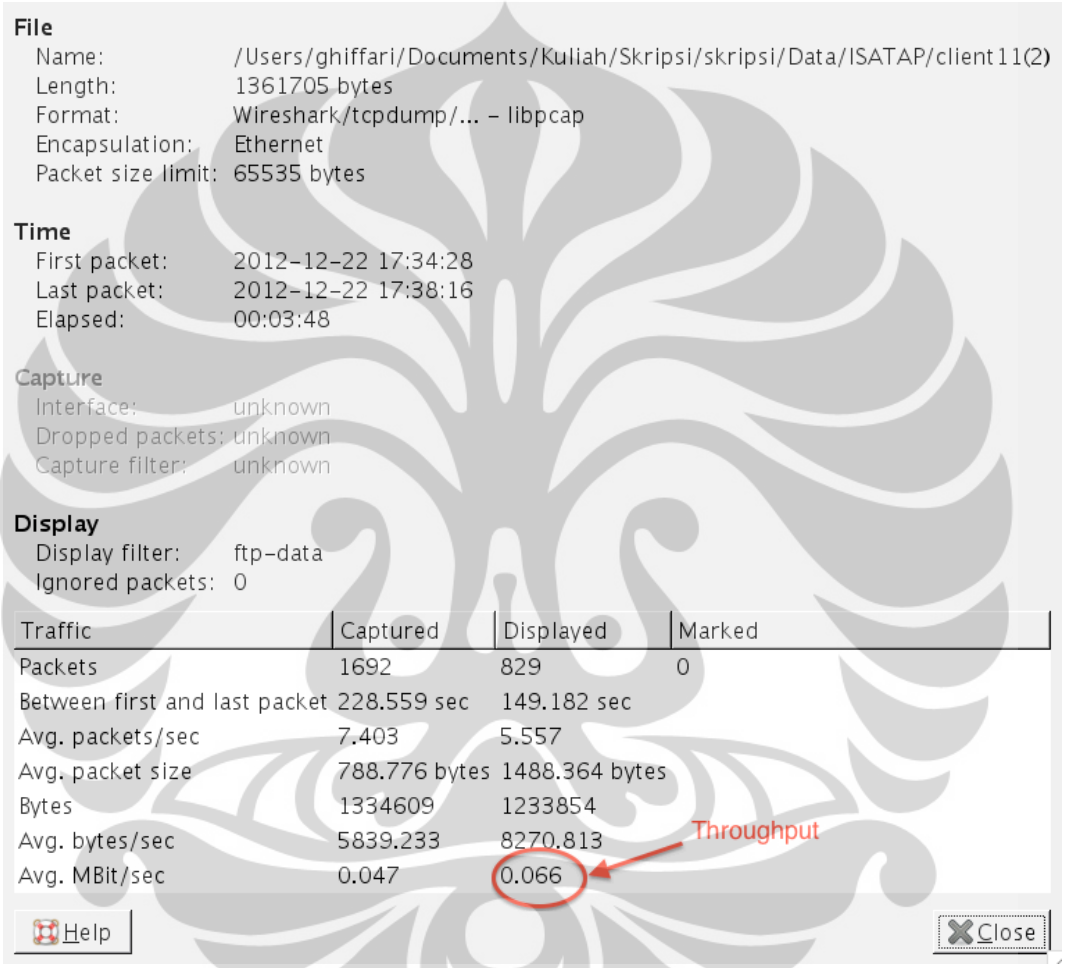
Nilai *throughput* didapatkan dengan formula:

$$Throughput = \frac{Jumlah\ paket\ yang\ dikirim}{total\ waktu\ pengiriman}\ packet/s$$

Akan tetapi, *throughput* tidak perlu dihitung dengan formula berikut pada data yang diambil karena paket yang ada berjumlah sangat banyak. Untuk itu, wireshark menyediakan fitur agar dapat mengetahui nilai *throughput*.

Informasi *throughput* didapatkan pada bagian *Statistics* dalam wireshark seperti Gambar 4.3. Akan tetapi, untuk mendapatkan *throughput* dari transfer file, paket harus disaring dahulu dengan parameter ftp-data agar didapatkan nilai *throughput* dari paket FTP-DATA saja.

Selanjutnya, seperti yang ditunjukkan Gambar 4.4, nilai *throughput* pada salah satu sampel data adalah 66 Kbps.



Gambar 4.4 Tampilan *throughput*

4.2.1. Pengukuran *Throughput*

Setelah dilakukan 10 kali percobaan, didapatlah data-data *throughput* pada pengiriman file melalui *tunneling* GRE dan *tunneling* ISATAP. Data yang didapatkan dirangkum pada Tabel 4.1.

Pada tabel terlihat data *throughput* pada kedua skenario yang menggunakan dua tipe *tunneling* yang berbeda. Karena pengambilan

sampel dilakukan 10 kali, maka terdapat 10 data *throughput* pada masing-masing skenario.

Tabel 4.1 Data *throughput*

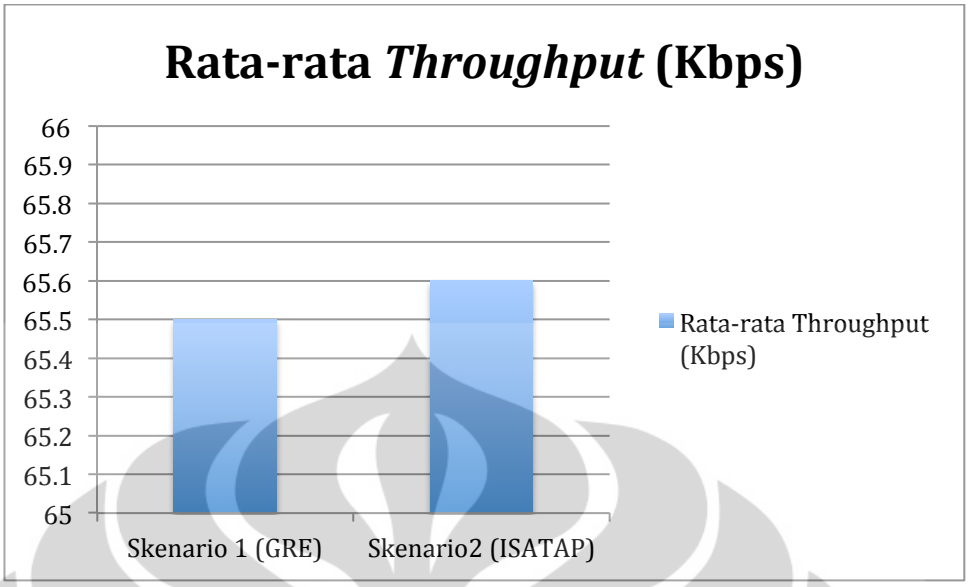
Pengujian ke-	*Throughput* (Kbps)	
	Skenario 1 (GRE)	Skenario 2 (ISATAP)
1	68	68
2	65	66
3	66	66
4	64	66
5	66	66
6	63	65
7	65	66
8	65	62
9	65	72
10	68	59
rata-rata	65.5	65.6

4.2.2. Analisis Throughput

Sebelum menganalisis data, *throughput* dirata-ratakan terlebih dahulu pada masing-masing skenario. Setelah itu, dibuat grafik rata-rata perbandingan kedua skenario. Grafik rata-rata *throughput* dapat dilihat pada Gambar 4.5.

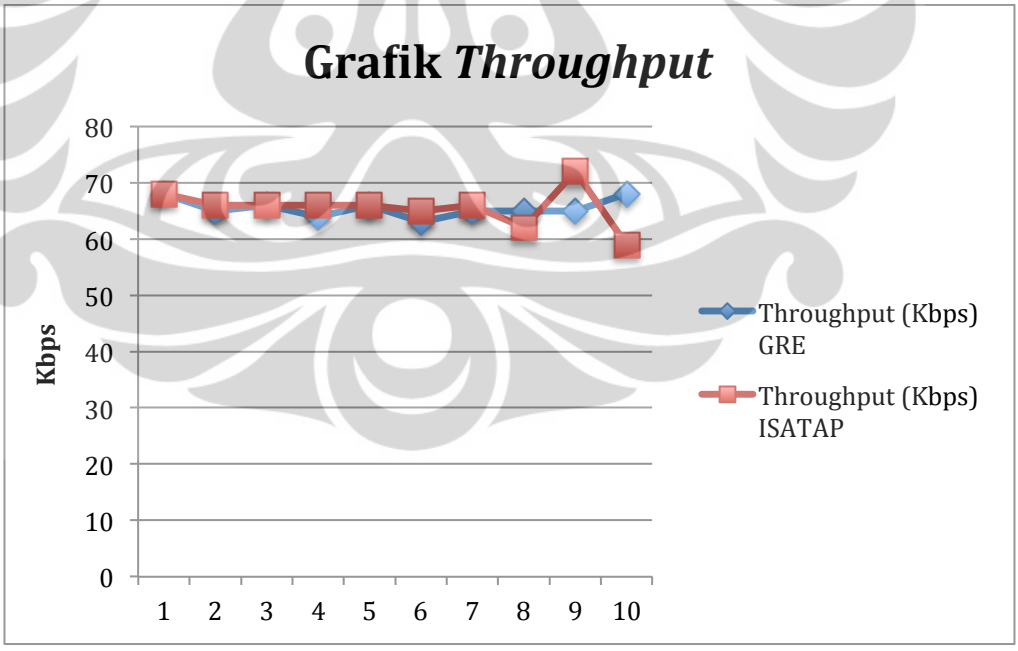
Gambar 4.5 menunjukkan perubahan rata-rata *throughput* untuk masing-masing skenario yang menggunakan tipe *tunneling* berbeda dan menggunakan file yang sama untuk pengiriman FTP. Pada skenario yang menggunakan *tunneling* GRE, *throughput* paling besar terjadi pada pengambilan sampel pertama dan terakhir, yaitu 68 Kbps. Sedangkan *throughput* terendah ada pada nilai 63 Kbps. Data *delay* mempunyai modus 65 Kbps dan rata-rata 65,5 Kbps.

Pada skenario yang menggunakan *tunneling* ISATAP, nilai terbesar adalah 72 Kbps pada pengambilan sampel ke-9 seperti terlihat pada Tabel 4.1. Nilai terkecil dari *throughput* adalah 59 Kbps dan modus berada pada nilai 65 Kbps serta rata-rata *throughput* adalah 65,6 Kbps.



Gambar 4.5 Grafik rata-rata *throughput* (Kbps)

Gambar 4.5 menunjukkan perubahan yang tidak signifikan antara kedua skenario yang menggunakan dua *tunneling* yang berbeda. *Throughput* FTP pada jaringan yang menggunakan *tunneling* ISATAP hanya berbeda 0,1 Kbps dari *tunneling* GRE.



Gambar 4.6 Grafik *throughput* pengiriman file

Hal ini juga ditunjukkan oleh Gambar 4.6 dimana pada 10 kali pengiriman, *throughput* pada kedua *tunneling* berada pada kisaran nilai yang sama yaitu 60-70 Kbps.

Pada proses pengujian, lebar pita pada kabel yang digunakan pada masing-masing PC (klien dan server) adalah 100 Mbps sedangkan untuk kabel serial adalah 1,544 Mbps. Setelah dilakukan pengujian, *throughput* tidak mencapai 1,544 Mbps karena *overhead* yang ada pada proses pembuatan *tunnel*.

Perbedaan *throughput* tidak terlalu besar karena *overhead* pada *header* yang digunakan pada GRE dan ISATAP tidak jauh berbeda. GRE menambahkan 24 byte *header*[10]. Sedangkan ISATAP menambahkan 20 byte header[9]. Hal ini yang menyebabkan kenaikan throughput pengiriman file menggunakan *tunneling* ISATAP (65,6 Kbps) hanya 0,15% dari throughput pengiriman file yang menggunakan *tunneling* GRE (65,5 Kbps).

4.3. Pengujian *Delay*

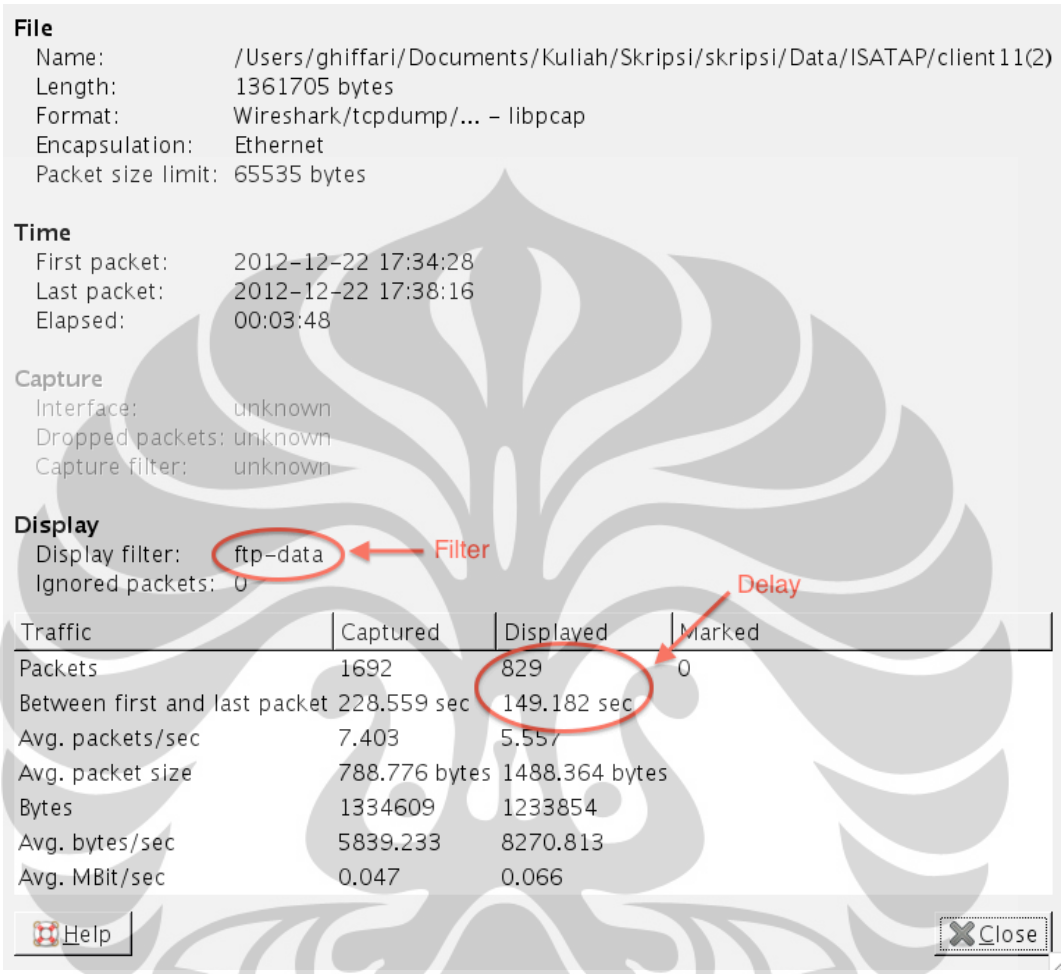
Delay merupakan waktu yang diperlukan paket untuk menempuh perjalanan dari *host* sumber ke *host* tujuan. Delay dapat dipengaruhi berbagai hal salah satunya adalah *latency*. *Latency* adalah waktu pemrosesan paket ketika melewati sebuah divais.

Latency biasanya hanya beberapa millisecond saja namun akan sangat mempengaruhi *delay* pada pengiriman paket jika terakumulasi karena banyaknya divais yang dilewati maupun karena waktu proses komputasi dari sebuah divais termasuk proses komputasi saat proses *tunneling*.

Karena paket yang ditangkap sangat banyak, akan sangat sulit jika harus menghitung manual. Wireshark menyedikan fitur untuk wireshark seperti pada Gambar 4.7.

Untuk memperoleh nilai delay dari wireshark, pilih *window* yang sama dengan *window* yang menunjukkan *throughput*. Perhatikan data waktu yang bertanda “Between first and last packet” yang mempunyai satuan detik. Delay

didapatkan dengan membagi data waktu pada “Between first and last packet” dengan jumlah paket yang ada seperti pada Gambar 4.7.



Gambar 4.7 Tampilan *delay*

4.3.1. Pengukuran *Delay*

Tabel 4.2 Data *delay*

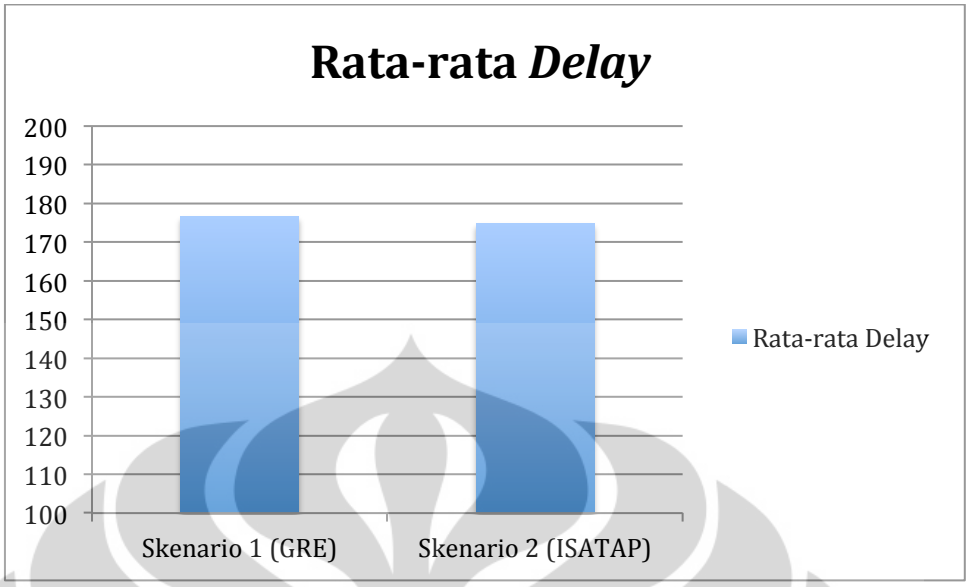
Pengujian ke	*Delay* (ms)	
	GRE	ISATAP
1	154.587	154.88
2	183.548	179.954
3	171.206	179.645
4	186.259	180.552
5	180.606	179.99
6	187.885	181.891
7	183.53	180.061
8	181.555	145.413
9	182.625	166.722
10	154.88	200.5
rata-rata	176.6681	174.9608

Sama seperti parameter sebelumnya, *throughput*, delay yang diambil sebagai sampel adalah 10 buah. Delay diukur di tiap skenario yang menggunakan *tunneling* berbeda.

Pada Tabel 4.2 terlihat data *delay* pada kedua skenario yang menggunakan dua tipe *tunneling* yang berbeda. Karena pengambilan sampel dilakukan 10 kali, maka terdapat 10 data *delay* pada masing-masing skenario.

4.3.2. Analisis *Delay*

Seperti pada analisis *throughput*, untuk menganalisis delay dibutuhkan nilai *delay* rata-rata dari 10 sampel. Setelah itu, dibuat grafik rata-rata perbandingan kedua skenario. Grafik rata-rata *throughput* dapat dilihat pada Gambar 4.8.

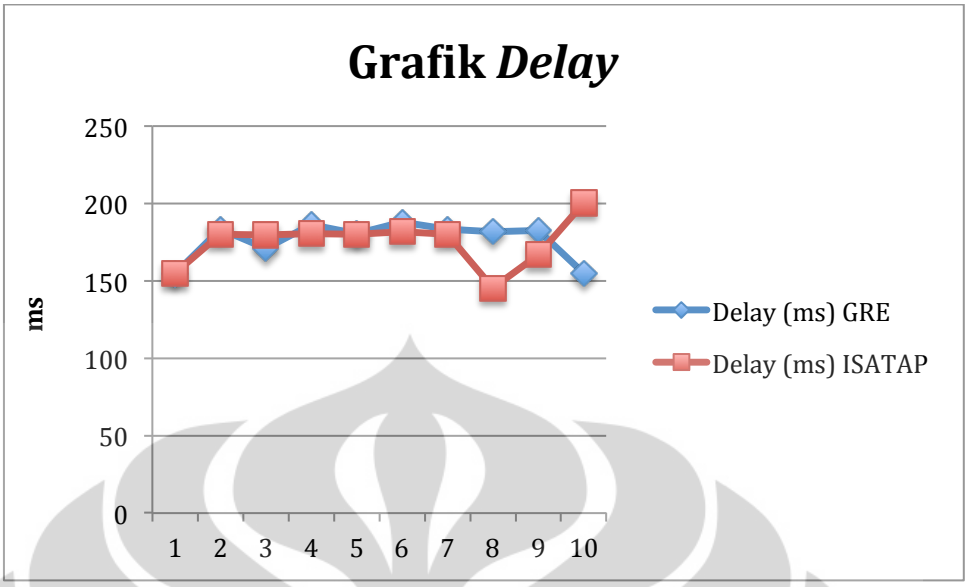


Gambar 4.8 Grafik rata-rata *delay*

Pada skenario yang menggunakan *tunneling* GRE, *delay* paling besar terjadi pada pengambilan sampel pertama, yaitu 187.885 ms. Sedangkan *delay* terendah terjadi pada pengambilan sampel keenam dengan nilai 154.587 ms.

Pada skenario yang menggunakan *tunneling* ISATAP, *delay* paling rendah terjadi pada pengambilan sampel kedelapan dengan nilai *delay* 145.413 ms dan *delay* tertinggi berada pada pengambilan sampel kesepuluh dengan nilai *delay* 200.5 ms. Sedangkan nilai tengah dari data *delay* didapatkan dengan membagi dua jumlah dari *delay* terurut kelima (179.954 detik) dan *delay* terurut keenam (179.99 ms), yaitu 179.972 ms.

Gambar 4.8 menunjukkan perubahan rata-rata *delay* untuk masing-masing skenario yang menggunakan tipe *tunneling* berbeda dan menggunakan file yang sama untuk pengiriman FTP. Rata-rata *delay* untuk skenario 1 (GRE) adalah 176.6681 ms dan rata-rata *delay* untuk skenario 2 (ISATAP) adalah 174.9608 ms.



Gambar 4.9 Grafik *delay* pengiriman file

Gambar 4.9 menunjukkan perbedaan yang tidak signifikan antara delay-delay yang didapatkan dari pengiriman file menggunakan kedua *tunnel*. Delay pada kedua skenario hanya berkisar 150-200 ms. Dilihat dari rata-rata *delay* kedua *tunnel* pada Gambar 4.8, *delay* pada GRE lebih besar 0.98% dibanding *delay* pada ISATAP.

Akan tetapi, *delay* akan terlihat jauh berbeda jika router yang dilewati oleh *tunnel* semakin banyak. Hal ini karena ISATAP adalah protokol berbasis TCP sedangkan GRE adalah protokol berbasis UDP. TCP yang berbasis *connection-oriented* akan selalu mengirim paket *acknowledgement* (ACK) setelah mengirim sebuah paket.

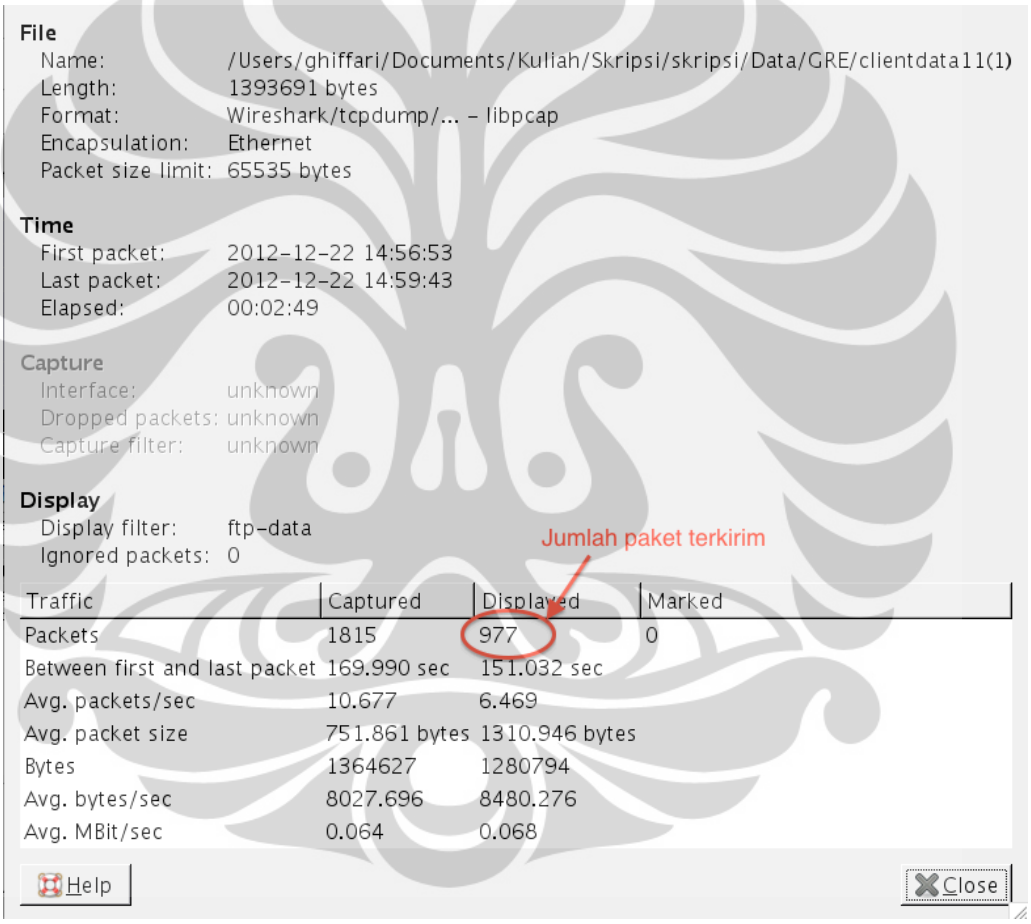
4.4. Pengujian Packet Loss

Packet loss merupakan salah satu parameter QoS yang merepresentasikan jumlah paket yang hilang saat pengiriman paket. *Packet loss* didapatkan dengan menghitung prosentase paket yang tidak sampai ke tujuan terhadap total jumlah paket yang sampai tujuan. Formula berikut menjelaskan perhitungannya:

$$Packet\ Loss = \frac{Jumlah\ paket\ dikirim - Jumlah\ paket\ diterima}{Jumlah\ paket\ diterima} \times 100\%$$

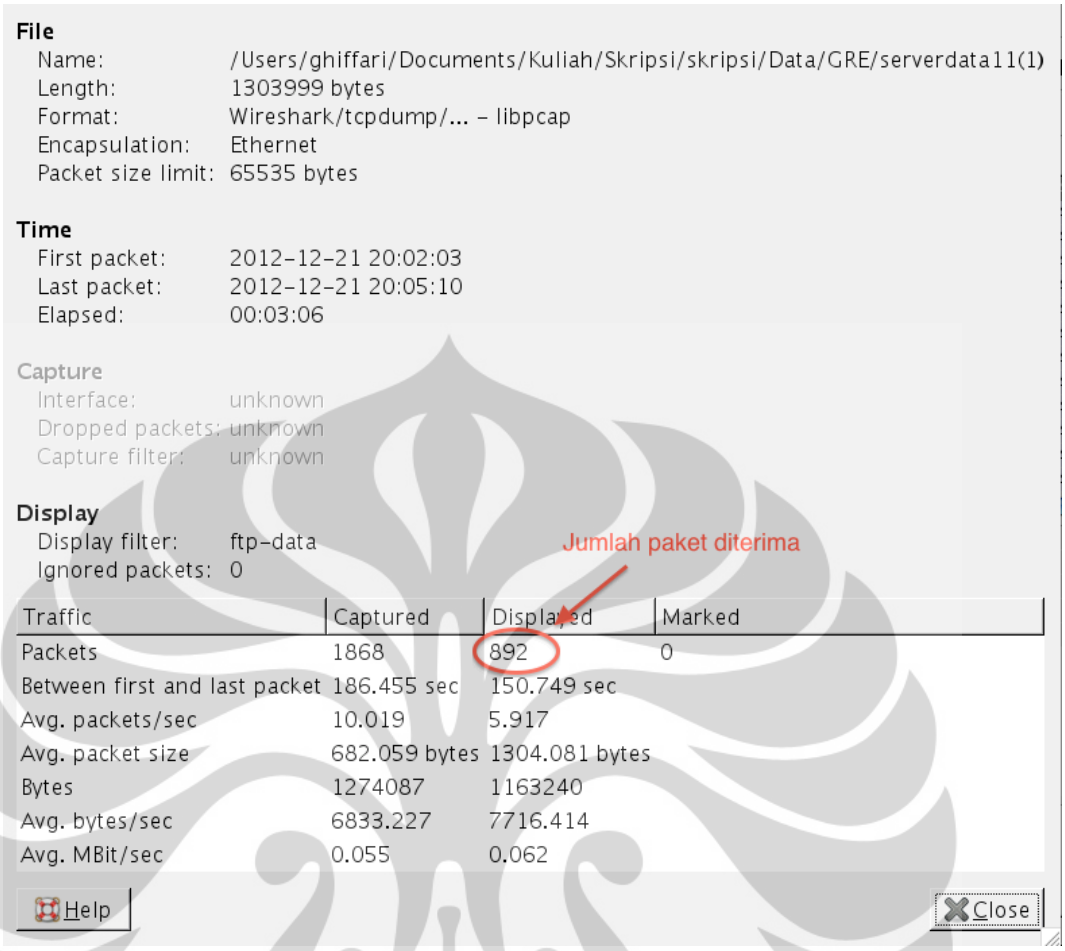
Paket loss adalah parameter yang sangat penting dalam menentukan QoS suatu layanan. Untuk itu, *packet loss* dapat menentukan tipe *tunneling* yang lebih efisien di antara *tunneling* GRE dan ISATAP.

Dalam wireshark, *packet loss* dapat dihitung dengan melihat jumlah paket di Summary pada data wireshark klien dan server. Untuk menghitung jumlah paket total FTP yang dikirim dan diterima, total paket yang ada harus disaring dengan perintah ftp-data sebelum melihat Summary pada wireshark. Gambar 4.10 dan 4.11 menunnjukkan jumlah paket yang dikirim oleh klien dan yang diterima oleh server.



Gambar 4.10 *Summary* client pada wireshark

Setelah jumlah paket tiap sampel diketahui, hitung prosentasenya dengan cara mengurangkan jumlah paket yang dikirim dengan jumlah paket yang diterima lalu dikalikan 100%.



Gambar 4.11 *Summary* server pada wireshark

4.4.1. Pengukuran Packet Loss

Sama halnya dengan pengukuran parameter-parameter sebelumnya, tiap skenario diambil 10 sampel sehingga akan didapatkan nilai statistik dari *packet loss* tiap *tunnel*. Tabel 4.3 menunjukkan data-data *packet loss* yang diambil.

Tabel 4.3 Data *packet loss*

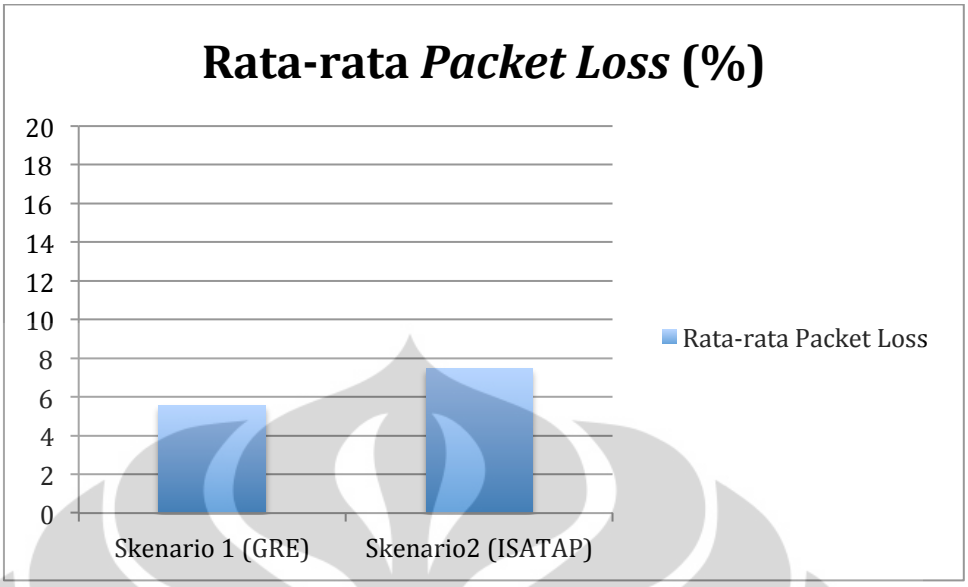
Pengujian ke	*Packet Loss* (%)	
	GRE	ISATAP
1	8.7	20.6
2	4.57	6.51
3	7.07	6.75
4	3.11	6.17
5	6.03	6.51
6	2.39	5.49
7	4.66	6.4
8	5.58	8.8
9	5.01	5.38
10	8.5	2.27
rata-rata	5.562	7.488

4.4.2. Analisis Packet Loss

Pada skenario yang menggunakan *tunneling* GRE, *packet loss* terendah terjadi pada pengambilan sampel keenam dengan besar *packet loss* 2.39%. Sedangkan *packet loss* tertinggi terjadi pada pengambilan sampel pertama dengan besar 8.7%. Modus *packet loss* pada skenario 1 didapat dengan menghitung rata-rata data terurut kelima (5.01%) dan keenam (5.58%), yaitu 5.295 %.

Pada skenario yang menggunakan *tunneling* ISATAP, *packet loss* terendah terjadi pada pengambilan sampel kesepuluh dengan besar *packet loss* 2.27%. Sedangkan *packet loss* tertinggi terjadi pada pengambilan sampel pertama dengan besar 20.6%. Modus *packet loss* pada skenario 1 didapat dengan menghitung rata-rata data terurut kelima (6.4%) dan keenam (6.51 %), yaitu 6.45 %.

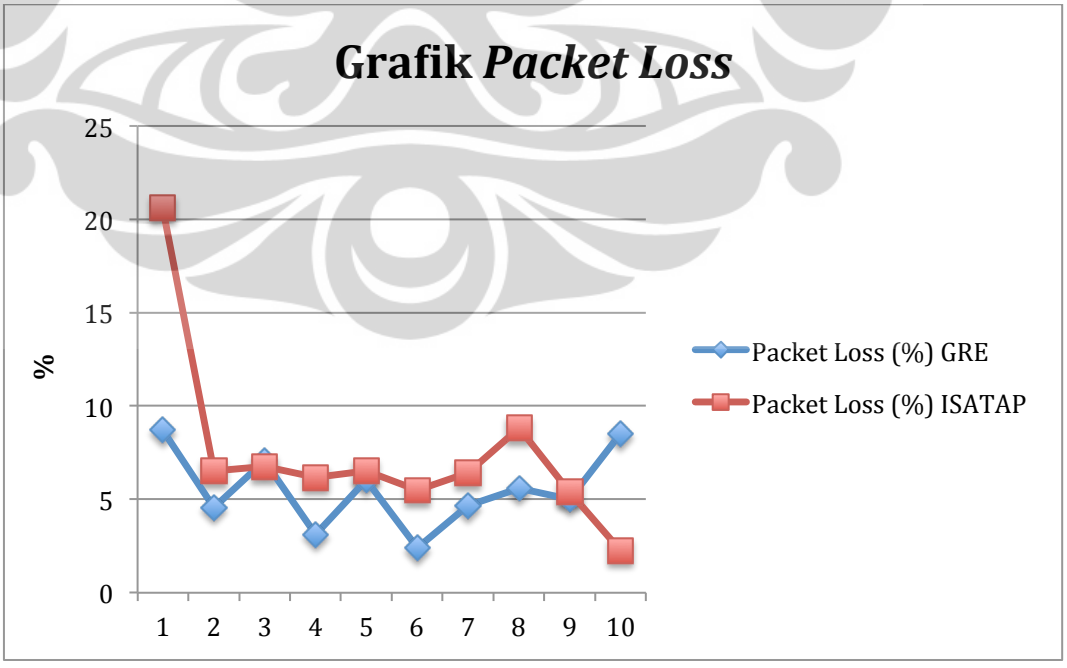
Rata-rata *packet loss* pada kedua *tunnel* ditunjukkan pada Gambar 4.12 dimana rata-rata *packet loss* skenario 2 lebih tinggi (7,488%) dari *packet loss* skenario 1 (5,562%). *Packet loss* pada pengiriman paket dengan *tunneling* ISATAP mempunyai kenaikan sebesar 34.63% dari *packet loss* dengan *tunneling* GRE.



Gambar 4.12 Grafik rata-rata *packet loss*

ISATAP yang merupakan *tunneling* yang bertipe dinamik memang memiliki *packet loss* yang lebih besar dibanding *tunneling* yang bersifat statik seperti GRE. Ini terjadi karena ISATAP memiliki mekanisme *buffering* dari *tunnel* sumber ke *tunnel* tujuan[7].

Terlihat pada Gambar 4.13, grafik *packet loss* pada skenario yang menggunakan *tunneling* ISATAP secara konstan berada di atas grafik *packet loss* pada skenario yang menggunakan *tunneling* GRE.



Gambar 4.13 Grafik *packet loss* pengiriman file

BAB V
KESIMPULAN

Dari hasil pengujian dengan melakukan pengukuran dan analisis terhadap performansi aplikasi FTP pada dua tipe *tunneling* (GRE dan ISATAP) dapat diambil kesimpulan bahwa:

1. Nilai *throughput* tidak terlalu berbeda antara jaringan yang menggunakan *tunneling* GRE dan ISATAP. Pada pengiriman file yang menggunakan *tunneling* ISATAP, *throughput* hanya mengalami kenaikan sebesar 0,15% dari *throughput* yang menggunakan *tunneling* GRE karena *overhead* pada *tunneling* GRE dan ISATAP juga tidak jauh berbeda. *Overhead* GRE berukuran 24 byte sedangkan *overhead* ISATAP berukuran 20 byte.
2. *Delay* juga hampir tidak berbeda antara jaringan yang menggunakan *tunneling* GRE dan ISATAP. Delay pada skenario yang menggunakan *tunneling* ISATAP hanya mengalami penurunan sebanyak 0,98% dari skenario yang menggunakan *tunneling* GRE. Perbedaan delay akan semakin besar pada *tunnel* yang melewati banyak router karena *tunnel* ISATAP dibentuk dengan menggunakan protokol *connection-oriented* TCP yang memerlukan proses *three-way handshake* pada setiap sesi pengiriman paket sedangkan *tunnel* GRE dibentuk dengan menggunakan protokol UDP yang tidak melakukan proses *handshaking*.
3. *Packet loss* pada kedua tipe *tunneling* jauh berbeda. Skenario yang menggunakan *tunneling* GRE mempunyai *packet loss* sebesar 5.562% sedangkan skenario yang menggunakan *tunneling* ISATAP mempunyai packet loss sebesar 7.488%. Perbedaan tersebut terjadi karena ada proses *buffering* pada *tunnel* ISATAP saat pengiriman paket.
4. Pada aplikasi transfer file, *packet loss* pada jaringan yang menggunakan *tunneling* ISATAP akan selalu lebih besar dari pada jaringan yang menggunakan *tunneling* GRE.

5. Secara umum, parameter-parameter QoS pada kedua tipe *tunneling* tidak jauh berbeda pada aplikasi FTP kecuali *packet loss*. ISATAP memiliki nilai parameter-parameter yang lebih baik dibandingkan GRE. Namun, *tunneling* GRE lebih baik digunakan pada aplikasi FTP jika koneksi antar router yang membentuk *tunnel* tidak stabil dan sering mengalami gangguan interferensi yang menyebabkan paket hilang saat pengiriman file.

DAFTAR REFERENSI

[1] Odom, Wendel. 2010. CCNP ROUTE 640-902 Official Certification Guide. Januari 2010. Halaman 537

[2] Jenis-jenis alamat IPv6. Diakses tanggal: 24 Desember 2012. http://giat501.wordpress.com/2011/06/04/jenis-jenis-alamat-ipv6/] Diakses pada 24 Desember 2012

[3] RFC 2080. Diakses pada tanggal: 25 Desember 2012.

[4] RFC 2740. Diakses pada tanggal: 25 Desember 2012.

[5] RFC 2784. Diakses pada tanggal: 25 Desember 2012.

[6] Multicast Routing and Forwarding Introduction. Diakses tanggal: 26 Desember 2012. http://www.h3c.com/portal/Products___Solutions/Technology/IP_Multicast/Technology_Introduction/200702/201355_57_0.htm

[7] RFC 5214. Diakses pada tanggal: 26 Desember 2012.

[8] Odom, Wendel. 2010. CCNP ROUTE 640-902 Official Certification Guide. Januari 2010. Halaman 635

[9] Se-Joon Yoon, Jong-Tak Park, Dae-In Choi, Hyun K. Kahng. Performance Comparison of 6to4, 6RD, and ISATAP Tunnelling Methods on Real Testbeds. International Journal on Internet and Distributed Computing Systems. Vol: 2 No: 2, 2012

[10] S P Maj, D Veal. Evaluating Layer 3 Device Tunneling and Access Control List Security Bandwidths Using B-Node Theory. IJCSNS International Journal of Computer Science and Network Security, Vol.10 No.9, September 2010

[11] R. Hassan, M. K. Sailan. End-to-End Baseline File Transfer Performance Testbed. Information Technology Journal. Vol.10 No.2, Asian Network for Scientific Information 2011.

[12] Odom, Wendel. 2010. CCNP ROUTE 640-902 Official Certification Guide. Januari 2010.

LAMPIRAN

Contoh konfigurasi router:

GRE

- Konfigurasi R1

```
Building configuration...

Current configuration : 1274 bytes
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R1
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
!
!
ip cef
!
!
ipv6 unicast-routing
!
!
!
!
!
!
interface Tunnel0
 no ip address
 ipv6 address 2001:0:0:1::1/64
 tunnel source Serial0/0/1
 tunnel destination 192.168.2.2
!
interface FastEthernet0/0
 no ip address
 duplex full
 ipv6 address 2001::1/64
!
```

```
interface Serial0/0/0
 no ip address
 shutdown
!
!
interface Serial0/0/1
 ip address 192.168.2.1 255.255.255.0
 clock rate 64000
!
!
no ip http server
no ip http secure-server
!
!
ipv6 route 2001:0:0:2::/64 Tunnel0
!
!
!
!
control-plane
!
!
!
!
!
!
gatekeeper
 shutdown
!
!
line con 0
 stopbits 1
line aux 0
line vty 0 4
!
!
end
```

ISATAP

- Konfigurasi R1

```
Building configuration...

Current configuration : 1274 bytes
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R1
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
!
!
ip cef
!
!
ipv6 unicast-routing
!
!
!
!
!
!
interface Tunnel0
 no ip address
 ipv6 address 2001:0:0:1::/64 eui-64
 tunnel source Serial0/0/1
 tunnel mode ipv6ip isatap
!
interface FastEthernet0/0
 no ip address
 duplex full
 ipv6 address 2001::1/64
!
```

```
interface Serial0/0/0
 no ip address
 shutdown
!
!
interface Serial0/0/1
 ip address 192.168.2.1 255.255.255.0
 clock rate 64000
!
!
no ip http server
no ip http secure-server
!
!
ipv6 route 2001:0:0:2::/64 2001::1:0:5FFE:C0A8:202
!
!
!
!
control-plane
!
!
!
!
!
!
gatekeeper
 shutdown
!
!
line con 0
 stopbits 1
line aux 0
line vty 0 4
!
!
end
```