UNIVERSITAS INDONESIA

Analisis Pengaruh Serangan Denial of Service terhadap Performansi Jaringan IPv6 yang Menggunakan Tunneling GRE dan Dual-Stack pada Aplikasi FTP

SKRIPSI

AZKA AULIA
0806459702

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

UNIVERSITAS INDONESIA

Analisis Pengaruh Serangan Denial of Service terhadap Performansi Jaringan IPv6 yang Menggunakan Tunneling GRE dan Dual-Stack pada Aplikasi FTP

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana

AZKA AULIA
0806459702

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	**: Azka Aulia**
NPM	**: 0806459702**
Tanda Tangan	**: ……**
Tanggal	**: 28 Desember 2012**

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama	: Azka Aulia
NPM	: 0806459702
Program Studi	: Teknik Komputer
Judul Skripsi	: Analisis Pengaruh Serangan Denial of Service terhadap Performansi Jaringan IPv6 yang Menggunakan Tunneling GRE dan Dual-Stack 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 Si (.....)

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

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

Ditetapkan di : Depok

Tanggal : 21 Januari 2013

KATA PENGANTAR

Puji serta syukur penulis panjatkan kehadirat Allah Subhanahu wa Ta'ala karena dengan limpahan berkat serta rahmat-Nya penulis dapat menyelesaikan skripsi ini. Penulisan skripsi/Tugas Akhir 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 Rabb semesta alam yang telah memberikan kekuatan kepada penulis untuk menyelesaikan skripsi ini.
2. Orang tua dan keluarga penulis.
3. Ir. Endang Sriningsih, MT Si. selaku dosen pembimbing skripsi yang telah meluangkan waktunya, serta masukan-masukan selama bimbingan.
4. Teman-teman sekelompok bimbingan seminar-skripsi dan teman-teman dari angkatan 2008 yang selama ini menjadi teman diskusi penulis.

Akhir kata, penulis berharap agar Allah Subhanahu Wa Ta'ala berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga laporan skripsi ini membawa manfaat.

Depok, 28 Desember 2012

Azka Aulia

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

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

Nama	: Azka Aulia
NPM	: 0806459702
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:

Analisis Pengaruh Serangan Denial of Service terhadap Jaringan Performansi IPv6 yang Menggunakan Tunneling GRE dan Dual-Stack 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 pemiliki Hak Cipta

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Tanggal : 28 Desember 2012

Yang menyatakan

(Azka Aulia)

ABSTRAK

Nama : Azka Aulia
Program Studi : Teknik Komputer
Judul : Analisis Pengaruh Serangan Denial of Service terhadap Jaringan IPv6 yang Menggunakan Tunneling GRE dan Dual-Stack pada Aplikasi FTP

Skripsi ini akan membahas pengaruh serangan Denial of Service pada performansi jaringan IPv6 yang menggunakan tunneling GRE dan *dual-stack*. Mekanisme transisi diperlukan untuk melakukan interkoneksi jaringan IPv6 ke tujuan melewati *core network* yang masih menggunakan IPv4 dengan metode *tunneling* GRE dan *dual-stack*. Jaringan berbasis IPv6 tersebut akan diserang dengan cara *flooding* yang ukuran paketnya meningkat. Dari keadaan tersebut, dapat diamati perubahan kinerja ketika serangan ditingkatkan. Parameter yang digunakan pada untuk penilaian performansi adalah *throughput*, *packet loss*, dan *delay*. Mekanisme transisi yang dibandingkan merupakan parameter penting untuk mengetahui kinerja jaringan IPv6 pada mekanisme transisi.

Thoughput pada jaringan *tunneling* GRE mengalami penurunan sebesar 9.27% dan pada jaringan *dual-stack* sebesar 17.62%. *Delay* tidak banyak berbeda, pada *tunneling* GRE *delay* lebih baik dari *dual-stack* sebesar 15.84%. Perbedaan yang signifikan terjadi pada *packet loss* dimana GRE mempunyai *packet loss* yang lebih besar, yaitu 8.36% dibandingkan *packet loss* pada *dual-stack* yang bernilai 6.74%. Oleh karena itu, *tunneling* GRE lebih baik digunakan pada aplikasi FTP dalam keadaan server diserang menggunakan *Denial of Service.*

Kata kunci: IPv6, *tunneling* GRE, *dual-stack,* Denial of Service, QoS

ABSTRACT

Name : Azka Aulia
Study Program : Computer Engineering
Title : Analysis of Impact of Denial of Service Attack to IPv6 Network Performance Using GRE Tunneling and Dual-stack on FTP Application

This thesis will discuss the effect of denial of service attacks on IPv6 network performance using GRE tunneling and dual-stack. Transition mechanisms needed to interconnect the IPv6 network to the destination through the core network is still using IPv4 with GRE tunneling method and dual-stack. IPv6-based networks will be attacked by flooding which package size increases. From these circumstances, changes in performance can be observed when the attack intensified. The parameters used for the assessment of performance is throughput, packet loss, and delay. Transition mechanism is an important parameter to determine the performance of IPv6 network transition mechanism.

GRE tunneling *throughput* on the network has decreased 9.27% and on dual-stack network 17.62%. Delay is not much different, the GRE tunneling delay is better than dual-stack by 15.84%. Significant differences occurred in which the GRE packet loss have a greater packet loss, which is 8.36% as compared to the dual-stack packet loss is 6.74%. Therefore, GRE tunneling is better used on FTP in attacked situation by Denial of Service.

Keyword: IPv6, GRE tunneling, dual-stack, Denial of Service, QoS

DAFTAR ISI

BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Perumusan Masalah 2
1.3 Tujuan Penelitian 2
1.4 Batasan Masalah 2
1.5 Sistematika penulisan 3
BAB II IPv6, METODE TRANSISI, DAN DENIAL OF SERVICE 4
2.1 IPv6 4
2.1.2 Kelebihan IPv6 dibanding IPv4 4
2.1.3 Format Header 5
2.1.4 Pengalamatan IPv6 7
2.1.5 Format Prefix 11
2.2 Tunneling dan Mekanisme Transisi 11
2.2.1 Mekanisme GRE 13
2.2.2 Metode Dual-stack 15
2.3 Denial of Service dan Keamanan IPv6 16
2.4 File Transfer Protocol (FTP) 17
BAB III IMPLEMENTASI JARINGAN DAN METODE PENGAMBILAN DATA 19
3.1 Uraian Perancangan 19
3.2 Implementasi 20
3.2.1 Spesifikasi Sistem 21
3.2.2 Topologi Jaringan IPv6 menggunakan GRE 24
3.2.3 Topologi Jaringan IPv6 menggunakan *dual-stack* 25
3.3 Metode Pengambilan Data 25
3.3.1 Skenario 1 27
3.3.2 Skenario 2 27
BAB IV PENGUKURAN DAN ANALISIS 28
4.1 Pengukuran Parameter QoS 28
4.2 Pengolahan data 28
4.3 Analisis Throughput 30
4.4 Analisis Packet Loss 33
4.5 Analisis Delay 34
4.6 Analisis Keseluruhan Pengaruh Serangan Denial of Service 37
BAB V KESIMPULAN 39
DAFTAR REFERENSI 40

DAFTAR GAMBAR

Gambar 2.1 ..... **6**
Gambar 2.2 ..... **12**
Gambar 2.3 ..... **13**
Gambar 2.4 ..... **14**
Gambar 2.5 ..... **18**
Gambar 3.1 ..... **20**
Gambar 3.2 ..... **21**
Gambar 3.3 ..... **22**
Gambar 3.4 ..... **23**
Gambar 3.5 ..... **24**
Gambar 3.6 ..... **24**
Gambar 3.7 ..... **25**
Gambar 4.1 ..... **29**
Gambar 4.2 ..... **29**
Gambar 4.3 ..... **30**
Gambar 4.4 ..... **31**
Gambar 4.5 ..... **32**
Gambar 4.6 ..... **33**
Gambar 4.7 ..... **34**
Gambar 4.8 ..... **36**
Gambar 4.9 ..... **36**

DAFTAR TABEL

Tabel 4.1 ..... **31**
Tabel 4.2 ..... **34**
Tabel 4.3 ..... **35**

DAFTAR LAMPIRAN

Konfigurasi R1 GRE **42**
Konfigurasi R1 *Dual-stack* **44**
Hasil *capture* Wireshark pada topologi GRE **46**
Hasil *capture* Wireshark pada topologi *Dual-stack* **47**

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kebutuhan terhadap Internet sudah tidak bisa dipisahkan dari kehidupan manusia sekarang. Jaringan internet yang begitu luas menyebabkan setiap orang di seluruh penjuru dunia dapat mengakses jaringan yang menggunakan Internet Protocol (IP). Bertambahnya pengguna Internet membuat alamat IPv4 (yang masih digunakan sekarang) menjadi semakin sedikit. IPv6 yang diusulkan para ahli di Internet Engineering Task Force (IETF) merupakan solusi yang paling mendekati saat ini. Meskipun begitu, IPv6 masih terus dikembangkan dan belum digunakan secara penuh. Salah satu isu penting yang menjadi perhatian adalah keamanan di jaringan IPv6. IPv6 atau *Internet Protocol version 6* adalah revisi terbaru dari Internet Protocol (IP), protokol komunikasi primer, yang diatasnya seluruh Internet terbangun. IPv6 telah dikembangkan oleh IETF sebagai antisipasi panjang masalah di IPv4 mengenai alamat yang akan habis.

IPv6 menggunakan 128-bit alamat, memungkinkan sampai 2^{128} atau lebih tepatnya 3.4×10^{38} alamat – lebih dari 7.9×10^{28} kali dibanding IPv4 yang menggunakan 32-bit alamat. IPv4 mengizinkan hanya 4,294,967,296 alamat yang unik di seluruh belahan dunia. Bagaimanapun, hal ini berarti dua protokol tidak cocok, kompleks ketika bertransisi ke IPv6. Alamat IPv6 umumnya ditampilkan ke *user* yang terdiri dari delapan grup dari 4 hexadesimal digit yang terpisah oleh titik dua (:). Selain dari kapasitas ketersediaan alamat IP, IPv6 memiliki beberapa keunggulan lain dibandingkan dengan IPv4 yaitu keamanan jaringan yang terintegrasi, kemampuan *multicasting, stateless configuration, header* yang lebih sederhana, mobilitas, dan adanya beberapa pilihan tambahan.

Keamanan Jaringan merupakan hal yang sudah terintegrasi ketika IPv6 didesain, termasuk opsi untuk IPsec atau IP Security. IPv6 tidak mengimplementasikan fitur interoperabilitas seperti pada IPv4, tetapi secara esensial membuat suatu jaringan paralel yang independen. Pergantian trafik antara dua jaringan memerlukan *gateway translator* yang spesial. Namun tidak

secara general disyaratkan karena umumnya sistem operasi pada komputer dan implementasi pada software baik transparansi protokol maupun transparansi akses ke dua jaringan salah satunya menggunakan protokol *tunneling* seperti 6to4, GRE, ISATAP atau Teredo.

Transisi dari IPv4 ke protokol IPv6 tidak akan cepat dan untuk beberapa periode kedua protokol tersebut masih dapat digunakan. Ada beberapa mekanisme transisi seperti *tunneling* dan konfigurasi *dual-stack* (didukung oleh IPv4 dan IPv6). Hal ini sangat penting bagi *network designer* dan *administrator* mengerti benar implikasi keamanan dari mekanisme transisi untuk diterapkan, seperti *firewall* dan sistem deteksi gangguan. Skripsi ini akan membahas pengaruh serangan dari celah keamanan di IPv6 dengan menggunakan *tunneling* GRE dan *dual-stack* serta serangan Denial of Service pada aplikasi FTP.

1.2 Perumusan Masalah

Pada skripsi ini, akan dianalisis suatu jaringan berbasis protokol IPv6 namun tetap menggunakan IPv4 untuk menghubungkan dua jaringan yang berbeda karena masih eksisnya penggunaan IPv4. Jaringan ini nantinya akan mengimplementasikan pengaturan mekanisme transisi dari IPv4 ke IPv6 dengan menggunakan *routing protocol* pada IPv6 serta mengetahui pengaruh serangan Denial of Service pada performa jaringan IPv6 sebelum digunakan secara menyeluruh. Masalah yang dibahas pada tulisan skripsi ini adalah:

1. Mekanisme transisi dari IPv4 ke IPv6 menggunakan *dual-stack* dan *tunneling* GRE
2. Implikasi keamanan dari mekanisme transisi yang digunakan dan pengaruhnya pada performansi jaringan berbasis IPv6

1.3 Tujuan Penelitian

1. Menganalisis pengaruh Serangan Denial of Service terhadap performansi Jaringan IPv6 yang menggunakan *tunneling* GRE dan *dual-stack*

1.4 Batasan Masalah

Pada skripsi ini, penelitian dilakukan mulai dari tahap perancangan jaringan IPv6 dengan menggunakan mekanisme transisi *dual-stack* dan *tunneling* GRE untuk menyesuaikan IPv4 yang masih dipakai. Pada skripsi ini akan dilakukan penelitian untuk melihat penurunan performansi akibat

bertambahnya ukuran paket serangan Denial of Service dari mekanisme transisi yang dipakai dengan aplikasi FTP. Dari banyaknya metode *tunneling* yang ada akan dipakai metode GRE dan *dual-stack* yang sering digunakan dan didukung oleh layanan pada divais Cisco.

1.5 Sistematika penulisan

Penulisan Tugas Akhir ini terdiri dari 4 bab dengan sistematika penulisan sebagai berikut:

BAB I PENDAHULUAN

Bab ini membahas tentang latar belakang, tujuan, pembatasan masalah, metodologi penelitian dan sistematika penulisan.

BAB II DASAR TEORI

Bab ini membahas dasar teori mengenai IPv6, mekanisme transisi menggunakan *dual-stack* dan *tunneling* GRE, serangan Denial of Service dan aplikasi FTP.

BAB III PERANCANGAN

Bab ini membahas mengenai rancangan dua kasus untuk melakukan serangan pada jaringan berbasis protokol IPv6 menggunakan mekanisme transisi yang berbeda dan terjadi transfer dokumen menggunakan aplikasi FTP.

BAB IV ANALISIS

Bab ini akan menganalisis pengaruh serangan Denial of Service terhadap performansi Jaringan IPv6 dan pengaruh penggunaan metode mekanisme transisi yang dipakai.

BAB V KESIMPULAN

Menjelaskan kesimpulan dari hasil perancangan dan analisis yang telah dilakukan.

BAB II

IPv6, METODE TRANSISI, DAN DENIAL OF SERVICE

2.1 IPv6

IPv6 merupakan kelanjutan dari IPv4 yang dikeluarkan oleh IETF. IPv4 pertama kali diluncurkan sekitar tahun 1970 untuk memfasilitasi komunikasi dan pertukaran informasi antara peneliti di pemerintahan dengan para peneliti di kalangan universitas. IPv6 didisain untuk melengkapi banyak hal yang belum ada di IPv4. IPv6 pertama kali dikembangkan tahun 1990 karena diprediksikan alamat yang ada akan habis seiring dengan perkembangan Internet dan *personal computer*. Hal-hal yang terjadi pada IPv4 diakumulasi selama 20 tahun untuk membuat IPv6 dapat dipakai di masa depan. IPv6 memiliki banyak keuntungan dibanding IPv4 karena alamat yang bisa digunakan jauh lebih banyak, header yang lebih sederhana dan efisien, serta *QoS* yang baik dan keamanan yang terintegrasi. IPv6 terus diteliti dan diperbaiki kelemahannya sehingga siap digunakan secara luas.

Pada tahun 1994 IPv6 pertama kali diperkenalkan di Toronto, Kanada dengan tujuan mengatasi kekurangan-kekurangan yang ada pada IPv4. IPv4 menyediakan pengalamatan dengan panjang 32 bit yang memungkinkan alamat sebanyak 2^{32} atau 4.294.967.296 alamat. Berbeda dengan IPv4, IPv6 memiliki panjang alamat 128 bit. Dengan panjang 128 bit, alamat yang mungkin tersedia adalah sebesar 2^{128} atau sebanyak 3,4 x 10^{38} alamat. Total alamat yang sangat besar ini bertujuan untuk menyediakan ruang alamat yang tidak akan habis untuk beberapa tahun ke depan, dan membentuk infrastruktur *routing* yang disusun secara rapih, sehingga mengurangi rumitnya proses *routing* dan *tabel routing*.[1] Karena kebutuhan akan suatu *Internet Protocol* baru itulah alasan dikembangkannya IPv6, versi penerus dari IPv4.

2.1.2 Kelebihan IPv6 dibanding IPv4

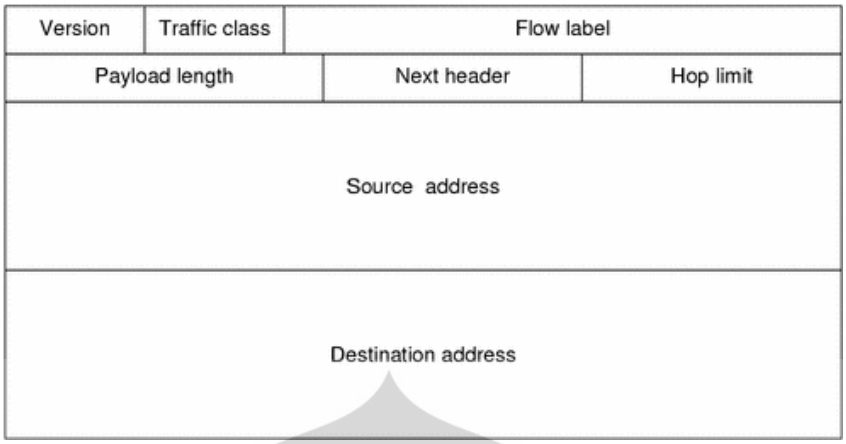
Sebagai evolusi dari IPv4, IPv6 memiliki banyak keunggulan yang merupakan penyempurnaan dari IPv4. Beberapa keunggulan tersebut diantaranya.

- Jumlah alamat yang tersedia pada IPv6 yang sangat besar dan dapat digunakan untuk beberapa tahun mendatang menggantikan IPv4.
- Format *header* yang baru

  Besar *header* IPv6 dua kali lebih besar dibandingkan IPv4. Namun, *header* pada IPv6 lebih sederhana dan lebih efisien.
- Infrastruktur *routing* yang rapi sehingga mengurangi rumitnya proses *routing* dan *table routing*.
- IPv6 menyediakan implementasi baru pada *multicast*
- Kemampuan *Plug-and-Play* Konfigurasi *host* dapat dilakukan secara *statefull* dengan menggunakan DHCPv6 dan juga secara *stateless* tanpa menggunakan DHCPv6
- Keberadaan *field low lable* pada *header* menunjukkan standar yang baik untuk mendukung *Quality of Service* (QoS) terutama untuk paket-paket yang bersifat *real-time*.
- Telah dikembangkannya protokol IPSec dan terintegrasi
- Mendukung sampai 1280-byte ukuran paket

2.1.3 Format Header

Perlu diketahui bahwa walaupun IPv6 address empat (4) kali lebih panjang dari IPv4 address, header IPv6 hanya dua kali dari panjang header IPv4. Oleh karena itu, IPv6 sangat mengurangi efek dari panjangnya kolom address. Kolom pada header IPv6 adalah: Version: nomor versi IP (4 bit). Kolom ini berisi nilai 6 untuk IPv6, dan nilai 4 untuk IPv4. Lokasi kolom ini sama untuk header IPv6 dan IPv4 sehingga memudahkan sebuah node untuk membedakan apakah ini paket IPv4 atau IPv6. Gambar 2.1 menjelaskan format *header* dari IPv6.



Gambar 2.1 *Header* pada IPv6 [2]

Priority memungkinkan sebuah sumber untuk mengidentifikasi prioritas pengiriman paket (4 bit). *Flow Label* digunakan oleh source untuk mengidentifikasi paket-paket dengan label tertentu ini membutuhkan teknik penanganan yang tertentu, seperti servis *real-time* antara sepasang host (24 bit). Payload Length: Panjang payload, bagian dari paket sesudah header, dalam oktet (16 bit). Nilai maksimum dari kolom ini adalah 65,535; jika kolom ini berisi nol mempunyai arti bahwa paket berisi payload yang lebih besar dari 64 Kbyte dan panjang payload yang sebenarnya ada di *Jumbo Payload hop-by-hop option*. *Next Header*: mengidentifikasi tipe header selanjutnya yang melekat pada header IPv6; menggunakan nilai yang sama dengan nilai pada IPv4 jika dimungkinkan (8 bit). Kolom *Next Header* dapat mengindikasikan option header, protokol pada lapisan yang lebih tinggi, atau tidak ada protokol di atas IP.

Hop Limit menspesifikasikan jumlah hop maksimum yang dapat dilalui sebelum paket di buang (8 bit). Nilai ini di set oleh *source* dan akan dikurangi 1 setiap kali melewati sebuah *node*. Paket akan dibuang jika nilai *Hop Limit* mencapai nilai nol. *Hop Limit* sama dengan Time To Live (TTL) di IPv4. Source Address: alamat IPv6 dari pengirim/ asal paket (128 bits). Destination Address: alamat IPv6 dari penerima paket (128 bits).

Contoh nilai dari Kolom Next Header:

Nilai Isi dari Next Header, 1 Internet Control Message Protocol (ICMP), 6 Transmission Control Protocol (TCP), 17 User Datagram Protocol (UDP), 43 Routing header, 44 Fragment header, 58 Internet Control Message Protocol version 6 (ICMPv6), 59 Tidak ada; ini adalah *header* yang terakhir 60 Destination Options header 89 Open Shortest Path First (OSPF).

2.1.4 Pengalamatan IPv6

Dalam arsitektur pengalamatannya alamat IPv6 mempunyai ukuran 128 bit yang artinya kira-kira berjumlah 2^{128} atau kira-kira 3,4 x 10^{38} alamat. Namun perhitungan teori ini tidaklah sepenuhnya akurat karena adanya hirarki *routing* dan kenyataan bahwa pada akhirnya nanti sebuah alamat akan didelegasikan sebagai blok yang bersambung dan bukan sebagai tiap-tiap satuan alamat.

Alamat IPv6 tersebut kira-kira akan terpotong setengahnya. Tidak akan pernah ada subnet yang memiliki 64 bit alamat signifikan atau lebih. Dari 128 bit tersebut hanya akan digunakan 64 bit untuk *routing* global dan internal yang disebut sebagai routing prefix. Sisa 64 bit dari alamatlah yang akan menunjukkan sebuah host pada suatu subnet yang disebut sebagai *host identifier* atau *host id*. Format alamat IPv6 adalah *colon-hexadecimal format*, berbeda dengan IPv4 yang menggunakan *dotted-decimal format*.

Alamat ini bisa direpresentasikan menjadi 8 segmen bilangan 16 bit dalam bilangan heksadesimal antara 0×0000 s/d 0xffff.

- Alamat asli

2001:d30:3:242:0000:0000:0000:1

- Penyederhanaan alamat tahap 1

2001:d30:3:242:0:0:0:1 Pada tahap ini terlihat bahwa blok-blok yang terdiri dari 4 (empat) angka 0, diringkas sehingga cukup hanya dituliskan dengan 1 (satu) angka 0 saja.

- Penyederhanaan alamat tahap 2

2001:d30:3:242::1 Pada tahap ini terlihat bahwa blok-blok yang memiliki angka 0 telah diwakilkan dengan tanda 2 *colon* (::), untuk mengetahui berapa banya blok yang diwakilkan maka dapat diketahui dengan cara 8 - (blok yang tertulis) = 8 - 5 = 3 blok atau 3x16 bit = 48

bit yang bernilai 0.

Untuk pendelegasian ke subnet biasanya akan dinyatakan dalam blok alamat yang dituliskan dalam blok alamat dengan panjang *prefix* tertentu dengan notasi CIDR menjadi 2001:d30:3:240::1/56.

Alamat IPv6 dapat diklasifikasikan menjadi 3 yaitu:

a. *Alamat Unicast*

Global Unicast, merupakan alamat dengan cakupan global dan unik sehingga bisa di-rute-kan di Internet. Selain global unicast, IPv6 juga mempunyai alamat local unicast dengan cakupan terbatas pada link lokal. Beberapa tipe alamat unicast IPv6 ini antara lain:

- Aggregatable global unicast addresses

Sering disebut sebagai alamat global, mirip dengan alamat publik pada IPv4 dan alamat ini ditandai dengan prefix 001. Alamat ini bisa dirutekan dan dijangkau secara global dari alamat IPv6 di Internet. Dinamakan *aggregatable* karena memang didesain untuk bisa diaggregasi dan diringkas (aggregation dan summarization) untuk menghasilkan infrastruktur routing yang efisien.

IANA telah mulai mengalokasikan blok alamat pertama untuk alamat global ini yaitu 2001::/16. Menurut kebijakan IANA setiap end-site seharusnya diberikan blok alamat IPv6 dengan panjang prefix /48.

- Link-local addresses

Alamat ini digunakan untuk berkomunikasi dalam cakupan link lokal yaitu pada link yang sama (misal jaringan flat tanpa *router*). *Router* tidak akan melewatkan trafik dari alamat-alamat ini keluar link. Alamat ini ditandai dengan prefix 1111 1110 10 atau FE80::/10. Alamat ini akan selalu diawali FE80 dan menggunakan *prefix* FE80::/64 dengan 64 bit selanjutnya adalah antarmuka id. Alamat link local ini dikonfigurasikan melalui IPv6 *autoconfiguration*.

- Site-local addresses

Alamat ini mirip dengan alamat privat pada IPv4 yang dalam teknologi IPv6 digunakan dalam cakupan *site* dan ditandai dengan

prefix 1111 1110 11 atau FEC0::/10. Alamat ini akan selalu diawali dengan FEC0. Karena sifatnya yang ambigu dan sulitnya pendefisinian baku dari cakupan site maka alamat ini dihapuskan penggunaanya.

- Special addresses

  Ada dua jenis alamat spesial pada IPv6 yaitu :

a. Alamat yang tidak dispesifikkan *(unspecified address*)

Sering disebut *all-zeros-address* karena memang bernilai 0:0:0:0:0:0:0:0 atau bisa dituliskan ::. Alamat ini sama dengan 0.0.0.0 di alamat IPv4. Alamat ini tidak boleh dikonfigurasikan pada antarmuka dan tidak boleh menjadi tujuan rute.

b. Alamat *loopback*

Jika alamat loopback pada IPv4 adalah 127.0.0.1 maka pada IPv6 dalah 0:0:0:0:0:0:0:1 atau bisa diringkas menjadi ::1. Alamat ini tidak boleh dikonfigurasikan pada antarmuka.

- Compatibility addresses

Alamat ini dibuat untuk mempermudah migrasi dan masa transisi dari IPv4 ke IPv6. Beberapa alamat ini antara lain :

a. Alamat IPv4-compatible
b. Alamat IPv4-mapped
c. Alamat 6over4
d. Alamat 6to4
e. Alamat ISATAP

- NSAP addresses

Adalah alamat yang digunakan untuk penterjemahan alamat Open System Interconnect (OSI) NSAP ke alamat IPv6. Alamat IPv6 ini ditandai dengan prefix 0000001 dan 121 sisanya adalah alamat NSAP.

b. Alamat Anycast

Alamat ini lebih menunjuk kepada fungsi layanan daripada alamat. Alamat anycast sama seperti alamat unicast IPv6 biasa (telah ditentukan dalam standar) dengan tambahan fitur bahwa router akan selalu merutekan ke tujuan yang terdekat atau lebih tepatnya terbaik sesuai yang telah dikonfigurasikan.

c. *Alamat Multicast*

Seperti halnya pada IPv4 pada IPv6 alamat ini menunjukkan sekumpulan piranti dalam grup multicast. Jadi alamat ini hanya akan muncul sebagai alamat tujuan, tidak akan pernah sebagai alamat asal. Jika paket dikirimkan ke alamat ini maka semua anggota grup akan memprosesnya.

Byte pertama menunjukkan bahwa ini adalah alamat multicast. Empat bit selanjutnya merupakan flag yang masing-masing telah didefinisikan. Bit pertama harus 0 karena dicadangkan untuk keperluan di masa mendatang. Bit kedua menunjukkan apakah alamat multicast ini mengandung alamat *Rendezvous Point* (RP), yaitu titik distribusi untuk aliran multicast tertentu dalam suatu jaringan multicast. Bit ketiga menandakan apakah alamat multicast ini mengandung informasi *prefix*. Sementara bit terakhir menunjukkan apakah alamat ini diberikan secara permanen.

Bagian berikutnya adalah cakupan (*scope*) yang digunakan untuk membatasi cakupan dari alamat multicast.

Alamat multicast ini memiliki sekup antara lain sebagai berikut :

Cakupan alamat multicast IPv6

Nilai cakupan	Deskripsi cakupan
0×0	Reserved
0×1	Node-Local
0×2	Link-Local
0×5	Site-Local
0×8	Organization Local
0xE	Global
0xF	Reserved

Bagian terakhir adalah penanda grup (Group ID). Pada prakteknya biasanya penanda grup ini dibatasi dalam 32 bit saja. Beberapa alamat multicast telah diberikan oleh IANA. Beberapa alamat yang diberikan ini dibuat untuk cakupan tetap dan beberapa diantaranya valid untuk semua cakupan.

2.1.5 Format Prefix

Dalam IPv4, sebuah alamat dalam notasi dotted-decimal format dapat direpresentasikan dengan menggunakan angka prefiks yang merujuk kepada subnet mask. IPv6 juga memiliki angka prefiks, tapi tidak digunakan untuk merujuk kepada subnet mask, karena memang IPv6 tidak mendukung subnet mask.

Prefiks adalah sebuah bagian dari alamat IP, di mana bit-bit memiliki nilai-nilai yang tetap atau bit-bit tersebut merupakan bagian dari sebuah rute atau subnet identifier. Prefiks dalam IPv6 direpesentasikan dengan cara yang sama seperti halnya prefiks alamat IPv4, yaitu [alamat]/[angka panjang prefiks]. Panjang prefiks menentukan jumlah bit terbesar paling kiri yang membuat prefiks subnet. Sebagai contoh, prefiks sebuah alamat IPv6 dapat direpresentasikan sebagai berikut:

3FFE:2900:D005:F28B::/64

Pada contoh di atas, 64 bit pertama dari alamat tersebut dianggap sebagai prefiks alamat, sementara 64 bit sisanya dianggap sebagai interface ID.

2.2 Tunneling dan Mekanisme Transisi

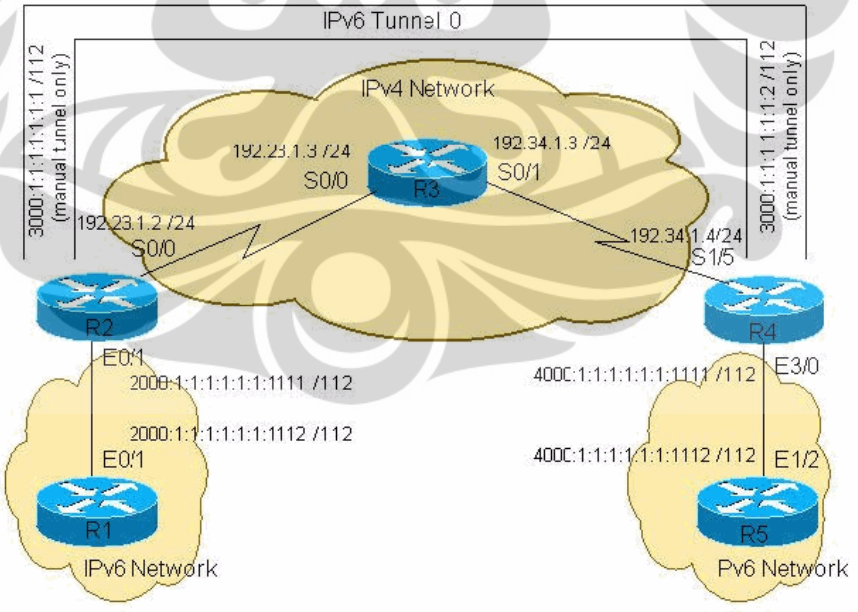
Ketika IPv6 diimplementasikan sekarang, jaringan yang menggunakan IPv6 tidak langsung terhubung antarjaringan. IPv6 mempunyai format alamat dan *header* yang berbeda dengan IPv4. Sehingga secara langsung IPv4 tidak bisa interkoneksi dengan IPv6 karena perbedaan fungsi pada beberapa bagian dan bagian yang baru pada IPv6.[15] Hal ini tentunya akan menimbulkan masalah pada implementasi IPv6 pada jaringan internet IPv4 yang masih eksis. Masalah tersebut adalah ketika sebuah *node* memiliki IPv6 dan untuk mengakses *node* tersebut lewat IPv4 dibutuhkan sebuah mekanisme yang dapat membuat *node* tersebut dapat diakses. Sebagai solusi masalah implementasi IPv6 ini diperlukan suatu mekanisme Transisi IPv6. Tujuan pembuatan mekanisme transisi ini adalah supaya paket IPv6 dapat dilewatkan pada jaringan IPv4 yang telah ada ataupun sebaliknya.

Teknologi Internet saat ini menggunakan protokol IPv4. Kenyataannya bahwa infrastruktur digunakan sekarang sangat menyulitkan transisi protokol dari IPv4 ke IPv6 sekaligus. Sehingga muncul banyak metode transisi yang

bisa digunakan diantaranya *Automatic tunneling*, *dual-stack*, dan *manually configured tunneling*. Mekanisme yang dibahas di sini adalah kondisi saat suatu node yang berbasis IPv6 harus berhubungan dengan *node* yang juga berbasis IPv6 dan menggunakan infrastruktur *routing* IPv4.

Dalam sebuah jaringan, protokol *tunneling* berimplikasi pada sebuah jaringan baru yang berfungsi sebagai jaringan pokok. Ada beberapa alasan penggunaan *tunneling*, yaitu untuk menyediakan jalur yang aman ketika paket melewati jaringan tak dikenal. Tunneling IPv6 memungkinkan masing-masing *host* dan *router* saling berkoneksi pada jaringan IPv4 yang masih digunakan. IPv6 mengenkapsulasi *datagram* dalam paket yang melewati jaringan IPv4. *Tunneling* disebut juga dengan enkapsulasi. Dengan metode ini, protokol IPv6 akan dienkapsulasi pada protokol IPv4. Paket yang terenkapsulasi ini kemudian diteruskan melalui jaringan IPv4. Proses enkapsulasi yaitu enkapsulasi pada *tunnel entry point*, dekapsulasi pada *tunnel exit point*, dan manajemen *tunnel*.

Bentuk mekanisme transisi ada berbagai macam diantaranya, *tunnel brokers,* 6to4, *dual-stack* (6in4), Teredo, ISATAP, GRE. Dalam skripsi ini, akan dibahas mengenai *tunneling* GRE dan metode *dual-stack*.



Gambar 2.2 Bentuk *tunneling* secara umum [3]

Gambar 2.2 menjelaskan bentuk *tunneling* pada empat jaringan IPv6 yang terhubung dengan sebuah *node* yang menggunakan alamat IPv4. Dua buah jaringan IPv6 yang terhubung langsung dengan *node* IPv4 memiliki

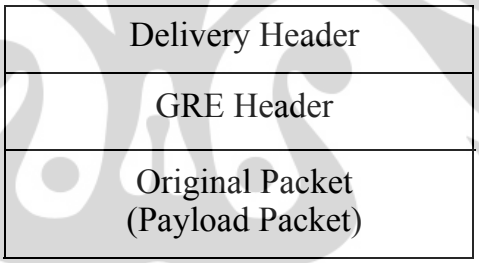
alamat IPv4 untuk berkomunikas. Alamat IPv4 pada dua buah jaringan IPv6 tersebut didapat dengan menggunakan mekanisme *tunneling*.

2.2.1 Mekanisme GRE

General Routing Encapsulation atau biasa disingkat GRE adalah sebuah metode tunneling yang dikembangkan oleh perusahaan jaringan Cisco Systems Inc. Metode GRE dapat mengenkapsulasi berbagai lebar dari protokol lapisan jaringan yang didalamnya terdapat *virtual point-to-point links* pada jaringan Internet.

Metode *tunneling* GRE dijelaskan pada RFC 2784. Secara umum, sebuah sistem mempunyai paket yang dienkapsulasi dan dikirim ke beberapa tujuan. Hal itu disebut *payload packet*. *Payload* adalah yang pertama kali dienkapsulasi di paket GRE. Hasil dari GRE bisa dienkapsulasi di beberapa protokol kemudian dikirim. Protokol yang diluar tersebut adalah protokol pengiriman.

Sebuah paket GRE yang dienkapsulasi mempunyai form:



Gambar 2.3 Sebuah paket menggunakan GRE [4]

Protokol tunneling yang satu ini memiliki kemampuan membawa lebih dari satu jenis protokol pengalamatan komunikasi. Bukan hanya paket beralamat IP saja yang dapat dibawanya, melainkan banyak paket protokol lain seperti CNLP, IPX, dan banyak lagi. Namun, semua itu dibungkus atau dienkapsulasi menjadi sebuah paket yang bersistem pengalamatan IP. Kemudian paket tersebut didistribusikan melalui sistem tunnel yang juga bekerja di atas protokol komunikasi IP.

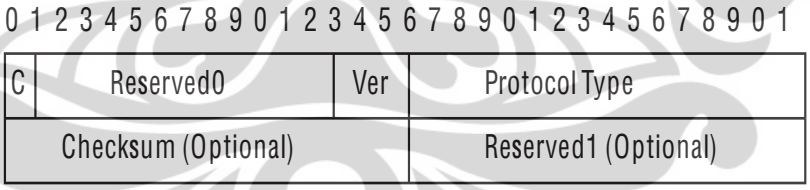
Dengan menggunakan tunneling GRE, router yang ada pada ujung-ujung tunnel melakukan enkapsulasi paket-paket protokol lain di dalam header dari protokol IP. Hal ini akan membuat paket-paket tadi dapat dibawa ke manapun dengan cara dan metode yang terdapat pada teknologi IP. Dengan adanya kemampuan ini, maka protokol-protokol

yang dibawa oleh paket IP tersebut dapat lebih bebas bergerak ke manapun lokasi yang dituju, asalkan terjangkau secara pengalamatan IP.

Aplikasi yang cukup banyak menggunakan bantuan protokol tunneling ini adalah menggabungkan jaringan-jaringan lokal yang terpisah secara jarak dan kembali dapat berkomunikasi. Atau dengan kata lain, GRE banyak digunakan untuk memperpanjang dan mengekspansi jaringan lokal yang dimiliki si penggunanya. Meski cukup banyak digunakan, GRE juga tidak menyediakan sistem enkripsi data yang lalu-lalang di tunnel-nya, sehingga semua aktivitas datanya dapat dimonitor menggunakan protocol analyzer biasa saja.

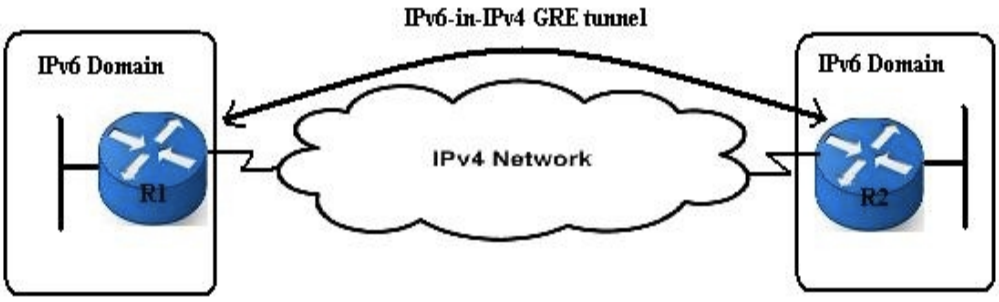
Dalam jaringan IPv4, GRE tunnel digunakan untuk menghubungkan IP *Private* suatu jaringan dengan IP *Private* jaringan lain yan melewati jaringan IP Public. Sedangkan pada IPv6 pada paket yang dikirimkan akan diberi *header* GRE tunnel dan IPv4. Inilah bedanya dengan IP in IP Encapsulation. Pada IPIP Encapsulation, *header* yang ditambahkan hanya berupa IP luar yang digunakan untuk berhubungan dengan jaringan Internet.

Paket *header* pada GRE tunnel memiliki struktur sebagai berikut:



Gambar 2.4 header pada GRE [5]

Sebenarnya adanya GRE *header* merupakan tambahan pada paket. Konsep ini mirip dengan mode IPSec tunnel. Paket asli dibawa melewati jaringan IP dan *header* paling luar digunakan untuk meneruskan sampai tujuan. Setelah paket GRE sampai pada tujuan akhir dari GRE tunnel, *header* eksternal akan dihapus dan paket internal akan diekspos lagi.



Gambar 2.5 Mekanisme *tunneling* GRE [6]

Gambar 2.5 menggambarkan koneksi antarjaringan yang menggunakan alamat IPv6 melewati jaringan yang berbasis IPv4 dengan mekanisme *tunneling* GRE. GRE memiliki beberapa karakteristik, secara umum karateristik tersebut adalah:

- GRE tunnel mirip dengan IPsec karena paket asli dibungkus didalam shell luar
- GRE merupakan *stateless*, dan tidak memberikan mekanisme kontrol aliran
- GRE menambah 24 bytes pada *overhead*, termasuk 20-byte IP *header* baru
- GRE multiprotocol dan dapat melakukan *tunnel* OSI Layer 3 apapun
- GRE mengizinkan *routing* untuk berjalan ke banyak *tunnel*
- GRE dapat membuat suatu paket bisa terus bertahan dalam trafik yang padat

2.2.2 Metode Dual-stack

Dual-stack mendukung *node* yang dapat menggunakan IPv4 dan IPv6 sekaligus. Cara termuda saat mekanisme transisi adalah satu divais dapat menyediakan IPv4 dan IPv6 sekaligus. Metode *dual-stack* menyediakan fasilitas tersebut sehingga dapat terkoneksi baik dengan divais yang mempunyai alamat IPv4 maupun IPv6. Dual-stack merupakan sebuah teknologi transisi IPv4 ke IPv6 yang fundamental yang melibatkan kekhasan dua implementasi protokol dalam satu sistem operasi. Metode *dual-stack* dijelaskan pada RFC 4213. Dengan

cara ini pemrogram dapat menulis aplikasi untuk menerima koneksi pada IPv4 maupun IPv6 atau bersifat *hybrid*.

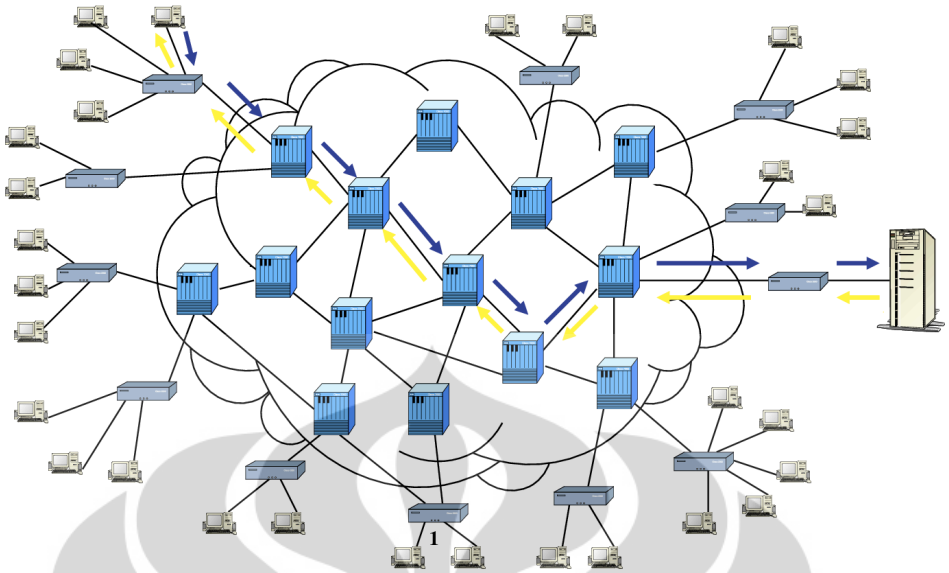
2.3 Denial of Service dan Keamanan IPv6

Keamanan jaringan menjadi bahan pembicaraan yang cukup banyak dalam teknologi jaringan. IPv6, yang merupakan IP *Next Generation* dari protokol sebelumnya yaitu IPv4, membenahi masalah keamanan dengan mengintegrasikan layanan keamanan berupa banyak fitur yang ada pada IPv6. Diantara fitur-fitur tersebut adalah *Authentication Header* (AH) dan *Encryption Security Payload* (ESP) serta fitur lainnya. Hal ini dilakukan agar berbagai serangan yang kemungkinan muncul bisa diatasi dengan fitur standar keamanan pada jaringan berbasis IPv6.

Untuk mencuri data, para penyerang menggunakan beberapa cara dalam bentuk serangan yang berbeda. Serangan yang umum dilakukan adalah serangan yang simultan (*reconnaissance attack*), Denial-of-Service, man-in-the-middle, *sniffing* (memata-matai), dan *spoofing* (pencurian paket). Serangan yang akan digunakan untuk menganalisis performansi dan kecenderungan pola serangan di IPv6 pada skripsi ini adalah serangan yang dapat melumpuhkan suatu layanan atau disebut Denial of Service

Para penyerang menggunakan Denial of Service untuk melumpuhkan suatu jaringan yang berupa Server sehingga layanan tersebut tidak bisa diakses dari luar. Sedangkan bentuk pengamanan jaringan IPv6 yang berkorelasi dengan *routing header* dijelaskan bahwa penyerang mampu menghindari *access control* melalui *routing headers*. Serangan berupa ICMPv6 dan yang berhubungan dengan alamat *multicast* tidak dapat diblok seperti IPv4 sehingga diketahuilah informasi berupa data-data dari jaringan yang diserang. [7]

Pengamanan suatu jaringan yang diserang dengan Distributed Denial of Service membutuhkan firewall atau suatu mekanisme agar serangan tersebut kembali lagi ke pengirim atau ke jaringan lainnya yang sering disebut *neighbor unreachable*.



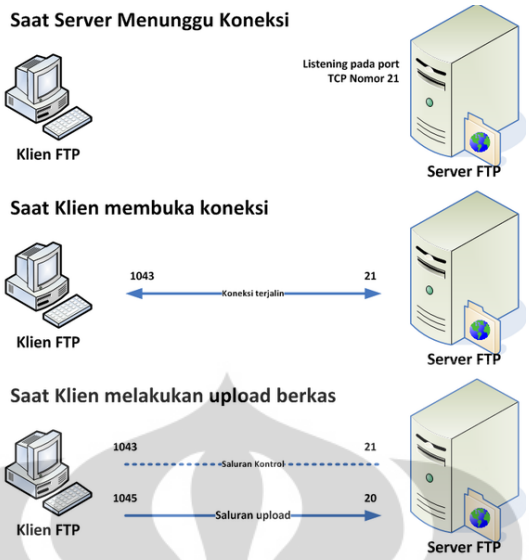
Gambar 2.6 Ilustrasi serangan Denial of Service [8]

Bentuk serangan melalui cara Denial of Service bermacam-macam diantaranya *flooding attack*, *smurf attack*, *SYN/ACK attack* dll. Pada skripsi ini, serangan yang dilakukan berupa *flooding* attack.

2.4 File Transfer Protocol (FTP)

File Transfer Protocol (FTP) atau protokol pengiriman berkas adalah sebuah protokol Internet yang berjalan di dalam lapisan aplikasi yang merupakan standar untuk pengiriman berkas (*file*) komputer antar mesin-mesin dalam sebuah komunikasi antarjaringan.

FTP merupakan salah satu protokol Internet yang paling awal dikembangkan, dan masih digunakan hingga saat ini untuk melakukan transfer *file* berkas-berkas komputer antara *client* FTP dan *server* FTP. Sebuah Klien FTP merupakan aplikasi yang dapat mengeluarkan perintah-perintah FTP ke sebuah server FTP, sementara server FTP adalah sebuah Service atau daemon yang berjalan di atas sebuah komputer yang merespons perintah-perintah dari sebuah klien FTP. Perintah-perintah FTP dapat digunakan untuk mengubah direktori, mengubah modus pengiriman antara biner dan ASCII, mengunggah berkas komputer ke server FTP, serta mengunduh berkas dari server FTP. Sebuah server FTP diakses dengan menggunakan *Universal Resource Identifier* (URI) dengan menggunakan format ftp://namaserver.



Gambar 2.5 Mekanisme FTP [9]

FTP menggunakan protokol *Transmission Control Protocol* (TCP) untuk komunikasi data antara *client* dan server, sehingga di antara kedua komponen tersebut akan dibuatlah sebuah sesi komunikasi sebelum pengiriman data dimulai. Sebelum membuat koneksi, TCP yang memiliki nomor port 21 di sisi server akan menerima percobaan koneksi dari sebuah *client* FTP dan kemudian akan digunakan sebagai port pengatur (*control port*) untuk (1) membuat sebuah koneksi antara klien dan server, (2) untuk mengizinkan klien untuk mengirimkan sebuah perintah FTP kepada server dan juga (3) mengembalikan respons *server* ke perintah tersebut. Sekali koneksi kontrol telah dibuat, maka server akan mulai membuka port TCP nomor 20 untuk membentuk sebuah koneksi baru dengan klien untuk mengirim data aktual yang sedang dipertukarkan saat melakukan pengunduhan dan penggugahan.

FTP hanya menggunakan metode autentikasi standar, yakni menggunakan *username* dan *password* yang dikirim dalam bentuk tidak terenkripsi. Pengguna terdaftar dapat menggunakan *username* dan *password* untuk mengakses, mengirim berkas-berkas yang ia kehendaki. Umumnya, para pengguna terdaftar memiliki akses penuh terhadap beberapa direktori, sehingga mereka dapat membuat berkas, membuat direktori, dan bahkan menghapus berkas. Pengguna yang belum terdaftar dapat juga menggunakan metode *anonymous login*, yakni dengan menggunakan nama pengguna anonymous dan *password* yang diisi dengan menggunakan alamat *e-mail*.

BAB III

IMPLEMENTASI JARINGAN DAN METODE PENGAMBILAN DATA

3.1 Uraian Perancangan

Dalam skripsi ini dibahas perancangan sebuah uji keamanan terhadap 2 *node* yang masing-masing memiliki *Local Area Network* (LAN) masing-masing. Dengan adanya fitur keamanan yang terintegrasi pada IPv6, celah-celah keamanan yang ada pada IPv4 dapat diatasi. Layanan keamanan IPv6 terus diuji sebelum dipakai seluruhnya.

Penelitian tentang keamanan banyak ditinjau dari serangan-serangan yang selama ini ada atau kemungkinan serangan yang akan muncul seperti *port scanning, brute force, denial-of-service, man-in-the-middle, sniffing* dan lain sebagainya.

Penelitian yang berhubungan dengan keamanan IPv6 selalu berkaitan dengan aspek-aspek berikut: pemantauan trafik, pemeriksaan paket, ICMPv6 dan multicast, Secure Neighbor Discovery protocol (SEND), CGAs dan fokus keamanan yang berhubungan dengan mekanisme transisi. Penyerang akan menggunakan serangan Denial of Service untuk menurunkan atau melumpuhkan kinerja pada sebuah jaringan.

Penelitian akan dilakukan dengan eksperimen pada laboratorium. Hal ini dilakukan karena lebih mudah mengatur lingkungan pada laboratorium daripada lingkungan jaringan yang sebenarnya atau simulasi. Bagaimanapun, eksperimen di laboratorium tidak akan memberikan hasil yang sebenarnya melainkan hanya aproksimasi saja. [10]

Perlu diketahui bahwa penelitian ini akan dilakukan berdasarkan data yang diambil dari serangan yang dilakukan yaitu berupa serangan Denial of Service. Uji ini dilakukan untuk menganalisis data-data pengaruh serangan Denial of Service terhadap kinerja IPv6 berdasarkan data-data yang didapat dari pemantauan trafik.

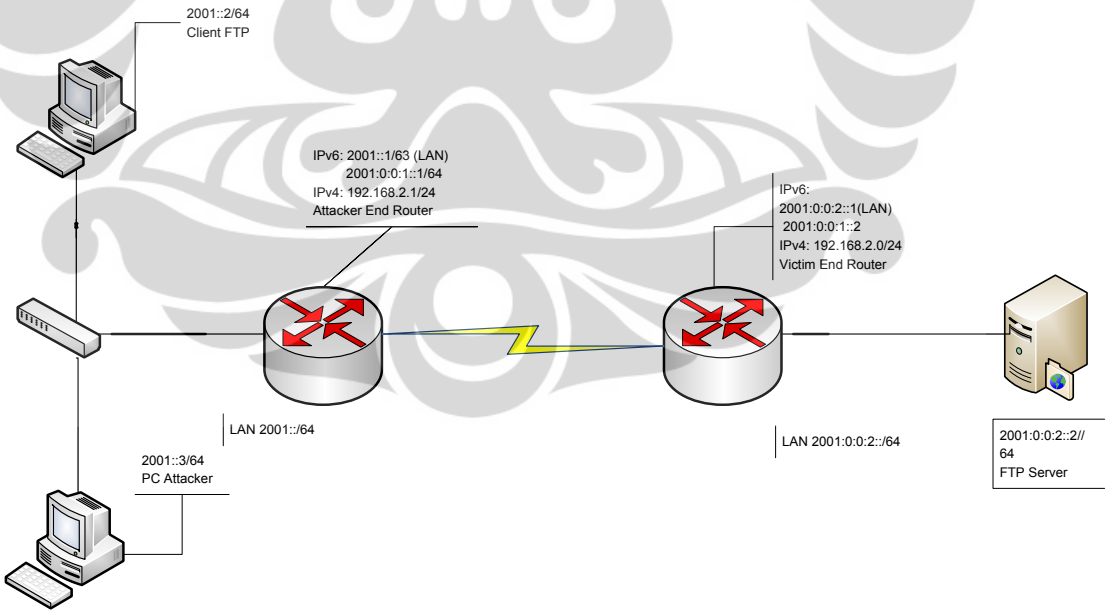
Pada perancangan topologi jaringan *test-bed* terdiri dari 2 tipe jaringan yang diklasifikasikan menurut teknik *tunneling* GRE dan *dual-stack*. Masing-masing jaringan akan menggunakan alamat IPv6.

Dua jaringan tersebut masing-masing direpresentasikan dengan *Attacker End* Router (AER) dan *Victim End* Router (VER). AER akan memiliki dua buah *host* yang memiliki fungsi berbeda, masing-masing sebagai penyerang server FTP pada VER dan *client* FTP. Pada VER, hanya menggunakan satu server yang difungsikan sebagai FTP Server. Lalu dari 2 jaringan tersebut akan dilakukan pengamatan mengenai kualitas kinerja *File Transfer Protocol* (FTP) yang diserang oleh serangan Denial of Service.

3.2 Implementasi

Perangkat pendukung utama dalam *test-bed* akan digunakan dua buah *router* produk dari Cisco Systems Inc. Alasan penggunaan *router* komersial karena *router* yang dipakai umumnya pada *real-network* adalah divais dari vendor yang memiliki *setup firewall* maupun protokol keamanan lainnya yang standar, sehingga mewakili keadaan sebenarnya. Dalam implementasinya, *router* tersebut menggunakan sistem operasi Cisco IOS untuk melakukan *routing* dan *tunneling* ke jaringan disebelahnya sehingga komunikasi dapat berjalan.

Secara global topologi jaringan yang digunakan untuk proses pengujian diperlihatkan pada Gambar 3.1.



Gambar 3.1 Topologi Jaringan secara global

Gambar 3.1 memperlihatkan topologi umum yang dipakai pada pengujian. Masing-masing LAN memiliki alamat IPv6 dan masing-

masing *gateway* memiliki alamat IPv6 dan IPv4. Attacker End Router (AER) adalah jaringan LAN tempat penyerang berada. FTP Server berada di jaringan Victim End Router (VER). Penyerangan akan dilakukan dari jaringan AER ke jaringan VER

3.2.1 Spesifikasi Sistem

Spesifikasi sistem yang digunakan terdiri dari perangkat keras dan perangkat lunak. Perangkat keras yang digunakan adalah sebagai berikut:

1. Attacker End Router (AER)

   Produk *Router* dari Vendor Cisco Systems Inc. series 1841 dengan Sistem Operasi Cisco IOS versi 12.4T, total RAM 256 MB, dan *Flash memory* 64 MB. *Router* ini digunakan sebagai AER didalam topologi ini. Gambar dari *router* tersebut bisa dilihat pada Gambar 3.2

Gambar 3.2 Cisco *router* 1800 series [12]

2. Victim End Router

   Produk *Router* dari Vendor Cisco Systems Inc. series 1841 dengan Sistem Operasi Cisco IOS versi 12.4T, total RAM 256 MB, dan *Flash memory* 64 MB. *Router* ini digunakan sebagai VER didalam topologi ini. *Router* ini sama seperti yang digunakan oleh AER.

3. PC Intruders

   PC dengan spesifikasi: Intel Centrino 2.4 GHz dan RAM 2 GB dengan Sistem Operasi Linux Ubuntu 12.10

4. PC Victims/ FTP Server

PC dengan spesifikasi: Intel Centrino 2.4 GHz dan RAM 2 GB dengan Sistem Operasi Linux Ubuntu 12.10

5. FTP Client

   PC dengan spesifikasi Intel Centrino 2.4 GHz dan RAM 2 GB dengan Sistem Operasi Linux Ubuntu 12.10

6. VER Switch

   Switch Cisco Catalyst 3560, satu-satunya *switch* yang dipakai dalam topologi ini. Gambar divais tersebut bisa dilihat pada Gambar 3.3

Gambar 3.3 Cisco Catalyst 3560 [13]

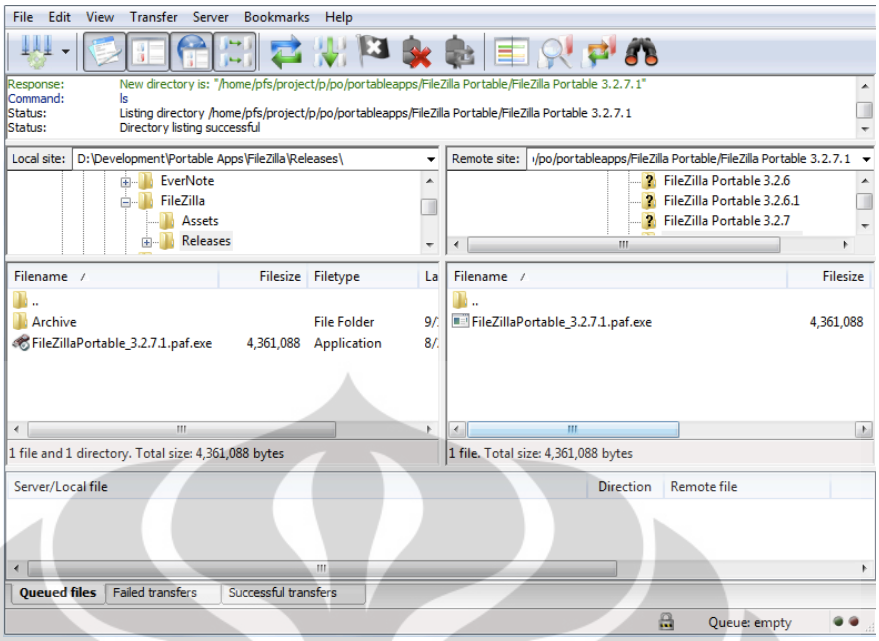
7. Kabel UTP, Serial, dan Console

   Kabel UTP yang dikonfigurasi dalam bentuk *cross* dan *straight*. Kabel dengan konfigurasi *cross* digunakan untuk menghubungkan *router2* ke PC Victim dan kabel *straight* digunakan untuk menghubungkan VER ke VER *switch* dan VER *switch* ke PC Attacker. Kabel Serial digunakan untuk menghubungkan antara VER dan AER serta kabel *console* untuk konfigurasi *router* dari PC.

Adapun perangkat lunak yang digunakan adalah sebagai berikut:

1. FileZilla

   FileZilla adalah perangkat lunak yang digunakan untuk melakukan transfer file dan mendukung fitur FTP.



Gambar 3.4 Antarmuka FileZilla[16]

2. Wireshark

   Wireshark adalah *software* untuk analisis jaringan dan protokol komunikasi yang bersifat *free* dan *open source*. Wireshark digunakan untuk menangkap paket yang masuk dan keluar suatu divais. Paket yang ditangkap dapat dianalisis dengan melihat isinya, contoh: alamat IP, nomor port, protokol aplikasi yang dipakai, dll

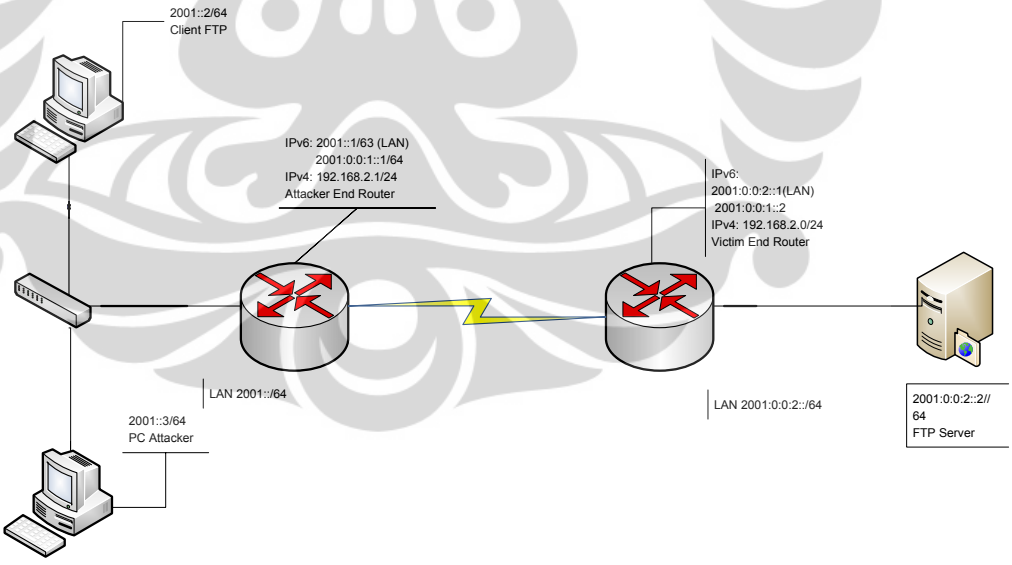
3. Backtrack 5

   Sistem Operasi berbasis Linux yang merupakan perubahan pada Sistem Operasi Debian yang dibuat khusus untuk melakukan serangan-serangan atau *hacking* dan mendukung berbagai fitur penyerangan mulai dari Denial of Service, *sniffing* sampai pencurian data.

Gambar 3.5 Sistem Operasi Backtrack 5 [14]

3.2.2 Topologi Jaringan IPv6 menggunakan GRE

Jaringan *test-bed* yang digunakan terdiri dari 2 macam topologi yaitu topologi jaringan IPv6 menggunakan GRE tunnel dan jaringan IPv6 menggunakan *dual-stack*. Gambar 3.5 adalah gambar topologi jaringan yang menggunakan mekanisme *tunneling* GRE (Generic Routing Encapsulation).



Gambar 3.6 Topologi Jaringan *tunneling* GRE

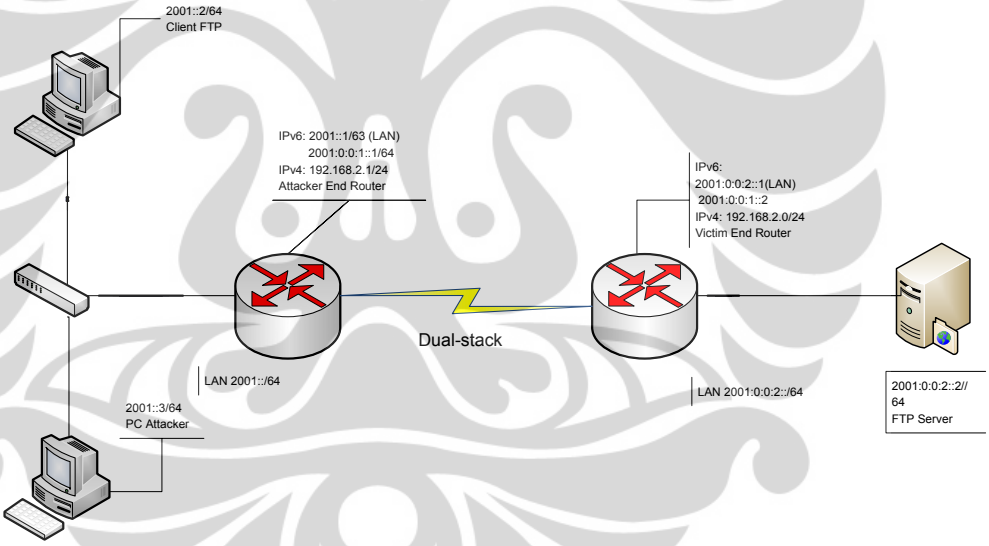
Pada topologi pertama semua komponen jaringan menggunakan alamat IPv6 dengan AER dan VER mempunyai alamat IPv4 dengan menggunakan *tunneling* GRE. *Routing protocol* yang

digunakan pada topologi ini adalah *routing static* karena hanya dua jaringan yang dipakai. Konfigurasi pada topologi ini bisa dilihat pada Lampiran.

3.2.3 Topologi Jaringan IPv6 menggunakan *dual-stack*

Begitu juga halnya dengan topologi sebelumnya, topologi kedua seluruh komponennya memiliki alamat IPv6 dan terdapat alamat IPv4 pada AER dan VER dengan menggunakan *dual-stack*. Pada topologi ini, *header* tidak ditambahkan ke paket yang dikirim melainkan terdapat dua alamat IP yaitu alamat IPv4 dan IPv6.

Alamat IPv4 dari Se0/0/0 dan Se0/0/1 adalah alamat tambahan yang diberikan lewat *dual-stack* sehingga dalam satu paket memiliki dua alamat IP. Bedanya dengan GRE adalah penambahan *header* pada paket yang membuat paket melewati jalur sampai tujuan direpresentasikan dengan alamat IPv4 saja.



Gambar 3.7 Topologi jaringan yang menggunakan *dual-stack*

Gambar 3.6 memperlihatkan topologi pada jaringan yang menggunakan metode *dual-stack*. Tidak ada banyak perbedaan dengan topologi jaringan GRE. Bedanya hanya pada mekanisme transisi yang digunakan.

3.3 Metode Pengambilan Data

Pada perancangan topologi jaringan IPv6 dirancang suatu jaringan yang mengakomodasi layanan keamanan yang telah teritegrasi pada IPv6 dan

perlindungan data pada *host* suatu jaringan. Jaringan yang akan dibangun menggunakan Cisco IOS Software yang sudah paripurna. Proses pengambilan data dilakukan dengan dua topologi dan dilakukan serangan dari AER ke FTP Server pada VER dalam bentuk *flooding* atau membanjiri Server dengan ICMPv6 dalam jumlah banyak.

Paket *flooding* yang dikirimkan dari PC Attacker dinamakan *flood*. Paket *flood* ini akan bervariasi yang rasionya berjumlah 100 bytes dengan pengiriman paket sebanyak enam kali pada masing-masing topologi dimulai dari 100 bytes sampai 600 bytes. Selama penyerangan dilakukan akan terjadi transfer dokumen berupa ekstension .pdf sebesar 1.5 MB dari *client* ke FTP server.

Pada pengujian ini tidak menggunakan format yang berbeda karena pengiriman dokumen tidak melihat format *file*. Setiap dokumen yang dikirim dipecah tiap 1500 bytes pada satu segmen. Dokumen yang digunakan berupa format PDF dengan jumlah tujuh halaman yaitu dokumen dengan nama '06111142.pdf'.

Dokumen berformat PDF ini akan dikirimkan dari *client* pada jaringan AER ke Server pada jaringan VER sebanyak enam kali untuk masing-masing topologi. Sehingga secara keseluruhan akan menghasilkan dua belas data untuk dianalisis. Wireshark akan merekam semua aktifitas jaringan termasuk paket yang keluar-masuk dan juga paket *flood* yang memenuhi server. Wireshark juga akan merekam *throughput. Transfer time* dan *delay* akan didapatkan dari hasil perhitungan. Pengambilan data dilakukan dengan melihat fitur *summary* pada Wireshark. Parameter *transfer time* didapat dari selisih waktu pengiriman *server* dengan jumlah paket yang diterima *client*. Setelah itu kemudian dibagi dengan waktu sampai paket yang dikirimkan server untuk dilihat persentasinya.

Parameter *throughput* didapat dengan melihat hasil pada wireshark dengan satuan Mbit/sec, bitrate, atau bytes/sec. Dari data yang didapat akan dibuat grafik untuk mengetahui *trend line*. Hasil pengambilan data digunakan untuk mengetahui kecenderungan penurunan performansi yang diakibatkan oleh serangan Denial of Service berupa *flooding* serta pengaruh penggunaan mekanisme transisi.

3.3.1 Skenario 1

Pengujian terhadap keamanan IPv6 secara umum dilakukan dalam beberapa tahap. Tahap pertama dua buah jaringan yang disiapkan masing-masing berfungsi sebagai penyerang dan korban, serta *client* dan server FTP. Tahap selanjutnya interkoneksi yang digunakan yaitu *tunneling* GRE yang memiliki konfigurasi tersendiri. Pada Cisco Router seri 1841 yang digunakan merupakan produk *router* yang telah memiliki fitur untuk melakukan koneksi yang cukup aman dan didukung layanan IPv6. Pada tahap pertama serangan dilakukan dalam bentuk *flooding* dari PC Attacker yang menggunakan sistem operasi Backtrack ke Server FTP sebanyak 100 bytes kemudian terjadi transfer dokumen sebesar 1.5 MB dari FTP *client* ke FTP Server. Pengambilan data dilakukan sebanyak enam kali dengan kenaikan paket *flood* 100 byte.

3.3.2 Skenario 2

Interkoneksi antarjaringan menggunakan *dual-stack* pada mekanisme transisi. Serangan dilakukan melalui PC Attacker yang menggunakan sistem operasi Backtrack pada AER ke FTP Server pada VER. Serangan dilakukan dalam bentuk *flooding*. Selama serangan dilakukan, terjadi transfer dokumen. Pengambilan data dilakukan sebanyak enam kali mulai dari serangan 100 bytes sampai 600 bytes dengan rasio 100 bytes. Mengacu ke [11], serangan Denial of Service memiliki pola dari hasil pengamatan setiap kenaikan ukuran paket sebesar 100 byte. Hal ini yang menjadi rujukan untuk mengukur performansi pada tiap kenaikan paket *flooding* 100 bytes. Hasil data yang didapatkan oleh wireshark akan dianalisis.

BAB IV

PENGUKURAN DAN ANALISIS

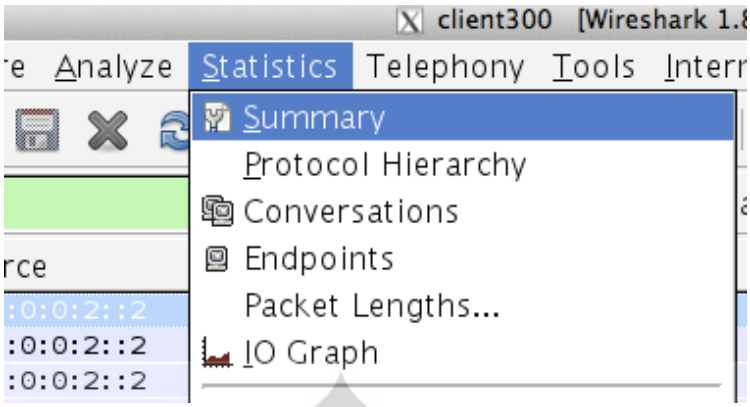
4.1 Pengukuran Parameter QoS

Berdasarkan topologi jaringan yang dijelaskan pada Bab III, pengujian akan dilakukan dengan keadaan Server yang diserang oleh Denial of Service sebanyak enam kali. Pengukuran pertama dilakukan dengan mengirimkan dokumen dari *client* ke FTP Server melewati jalur yang menggunakan *tunneling* GRE dengan alamat IPv6 yaitu, 2001::2/64 pada *client*, 2001:0:0:2::2/64 pada Server. AER memiliki alamat 2001::1/64 untuk IPv6 dan 192.168.2.1 dengan subnet mask 255.255.255.0 untuk IPv6. Sedangkan VER memiliki alamat 2001:0:0:2::1/64 untuk IPv6 dan 192.168.2.2 dengan subnet mask 255.255.255.0 untuk IPv4

Kenaikan 100 bytes dipilih karena pola serangan Denial of Service terbentuk setiap kenaikan ukuran paket sebesar 100 bytes dengan *rate* yang terus meningkat berdasarkan hasil penelitan. [11] Pengukuran kedua hampir sama dengan pengukuran pertama namun dibedakan dalam metode transisi yang digunakan yaitu *dual-stack*. Pengukuran penurunan performa aplikasi FTP ini diukur dengan *throughput*, *packet loss*, dan *delay.*

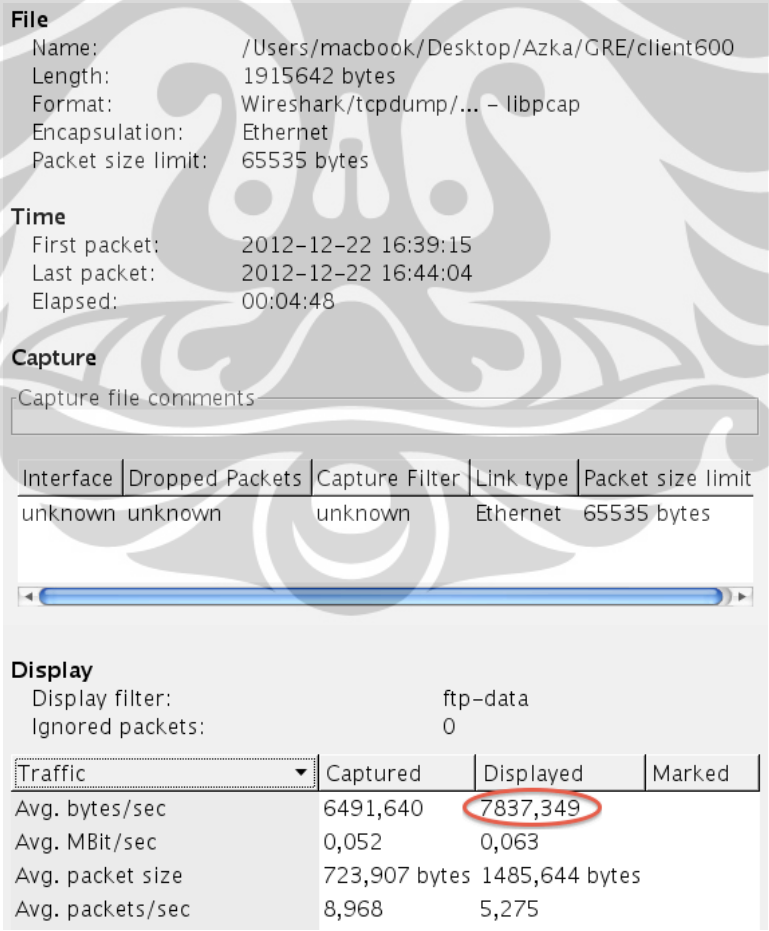
4.2 Pengolahan data

Perhitungan parameter menggunakan *throughput*, *packet loss*, dan *delay* berbeda-beda. Nilai pada parameter tersebut diolah dari data yang didapat oleh wireshark. Sebagaimana diketahui bahwa throughput merupakan kecepatan rata-rata pada paket dalam satuan bitrate, Mbit/sec atau MB/sec. Pengolahan data *throughput* pada skripsi ini menggunakan Mbit/sec. Data tersebut diambil dari Wireshark dengan memilih Toolbar ‘Statistic’ kemudian pilih Summary. Hal ini seperti diperlihatkan pada Gambar 4.1



Gambar 4.1 Menu Summary pada Wireshark

Sebelumnya dilakukan filtering data dengan kata 'ftp-data'. Pada Summary tersebut terdapat Average bytes/sec dan nilai tersebut yang akan digunakan untuk menghitung kecenderungan penururan nilai *throughput*. Pada Gambar 4.2 diperlihatkan hasil Summary pada wireshark.



Gambar 4.2 Nilai *throughput* pada Wireshark

Nilai yang dilingkari dengan warna merah merupakan nilai *throughput* dari salah satu sampel data yang didapatkan dari hasil pengukuran. Satuan yang dipakai dalam *throughput* pada pengukuran ini adalah bytes/sec.

Untuk Packet Loss, perhitungan didapatkan dari sebuah rumus:

$$\frac{Paket\ yang\ dikirim - Paket\ yang\ diterima}{Paket\ yang\ dikirim} \times 100\%$$

Paket yang dikirim merupakan paket yang diunggah oleh FTP *client* ke Server. Paket tersebut dapat dilihat pada data wireshark yang direkam di PC Client FTP. Sedangkan paket yang diterima adalah paket yang diterima oleh FTP Server dari FTP *client*. Jumlah paket tersebut dapat dilihat pada Summary di Wireshark.

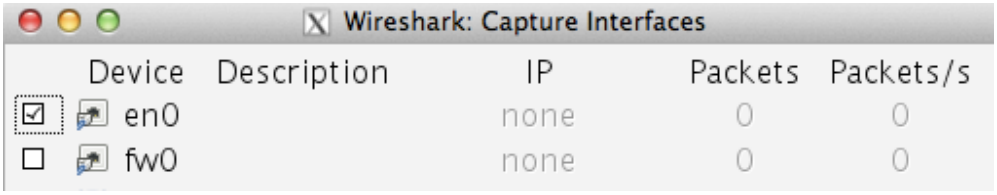
Untuk menghitung *delay*, pada Gambar 4.3 dapat dilihat terdapat nilai 'between first and last packet'. Nilai tersebut dibagi dengan jumlah paket pada kolom 'Displayed' untuk diketahui *delay* pada pengiriman file.

Traffic	Captured	Displayed
Packets	1936	889
Between first and last packet	207,441 sec	172,750 sec
Avg. packets/sec	9,333	5,146

Gambar 4.3 Parameter untuk menghitung *delay*

4.3 Analisis Throughput

Throughput adalah kecepatan transfer rata-rata yang mampu ditransfer melewati jaringan setiap satuan waktu. Throughput dapat dinyatakan dalam Packet/second, byte/second, atau bits/second. Pengambilan parameter transfer time dilakukan dengan cara mengunduh dokumen dari FTP Server ke FTP Client. Sebelumnya FTP Server diserang dalam bentuk *flooding* oleh PC Attacker. Pada saat sebelum diserang, wireshark telah diaktifkan untuk merekam aktifitas pada jaringan dalam bentuk paket-paket yang berbeda. Wireshark digunakan untuk merekam aktifitas antarmuka di en0.



Gambar 4.4 Antarmuka en0 yang direkam oleh Wireshark

Pada Gambar 4.4 diperlihatkan antarmuka en0 yang direkam aktifitasnya oleh wireshark yang terhubung ke *gateway*.

Karena paket-paket yang terekam oleh wireshark sangat banyak maka dilakukan filtering untuk memfokuskan data pada paket FTP. Filtering yang dilakukan dengan menuliskan kata 'ftp-data' khusus pada paket yang dikirimkan dalam protokol FTP. Selain ftp-data, paket-paket pada aplikasi FTP untuk komunikasi antar *node* juga terekam oleh wireshark

Berdasarkan hasil pengukuran, terdapat kecenderungan bahwa nilai *throughput* semakin kecil untuk setiap penambahan serangan DoS. Data tersebut dapat dilihat pada tabel dibawah. Tabel 4.1 dibawah ini merupakan nilai *throughput* yang didapat dari hasil pengukuran dalam satuan Kbytes/sec.

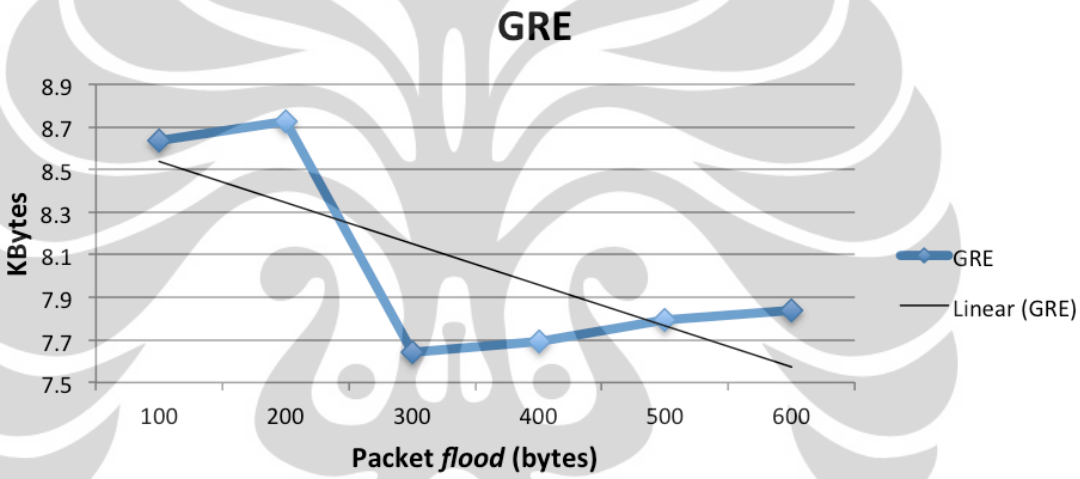
Tabel 4.1 Nilai *throughput*

Ukuran Paket (bytes)	GRE (Kbytes)	Dual-stack (Kbytes)
100	8.6386	8.3234
200	8.7272	8.4666
300	7.6419	7.9086
400	7.6932	8.1477
500	7.7943	7.9646
600	7.8373	6.8560

Dari hasil pengukuran didapat enam data untuk masing-masing topologi yang nilainya cenderung menurun. Nilai *throughput* tertinggi pada topologi GRE sebesar 8.6386 Kbytes/sec dan yang terendah sebesar 7.6419 Kbytes/sec. M serta memiliki median sebesar 7.6676 Kbytes/sec. Pada topologi *dual-stack* nilai tertinggi sebesar 8.4666 Kbytes/sec dan yang terendah sebesar 6.8560 Kbytes/sec serta median sebesar 8.02815 Kbytes/sec.

Ukuran paket merupakan serangan Denial of Service yang bertahap mulai dari 100 bytes hingga 600 bytes. Kolom GRE dan Dual-stack menunjukkan nilai *throughput* yang didapat ketika terjadi transfer dokumen dari *client* ke FTP Server. Untuk serangan diatas 600 bytes pada dual-stack tidak berhasil karena dokumen yang ditransfer tidak dapat sampai ke FTP Server dengan utuh. Pengiriman dokumen diatas 600 bytes selalu berhenti ketika transfer file sekitar 800 KB. Hasil yang sama didapat untuk pengujian sampai tiga kali serangan sehingga pengukuran dibatasi sampai serangan sebesar 600 bytes.

Hasil yang menunjukkan nilai throughput pada masing-masing pengujian ditampilkan pada grafik dibawah ini:

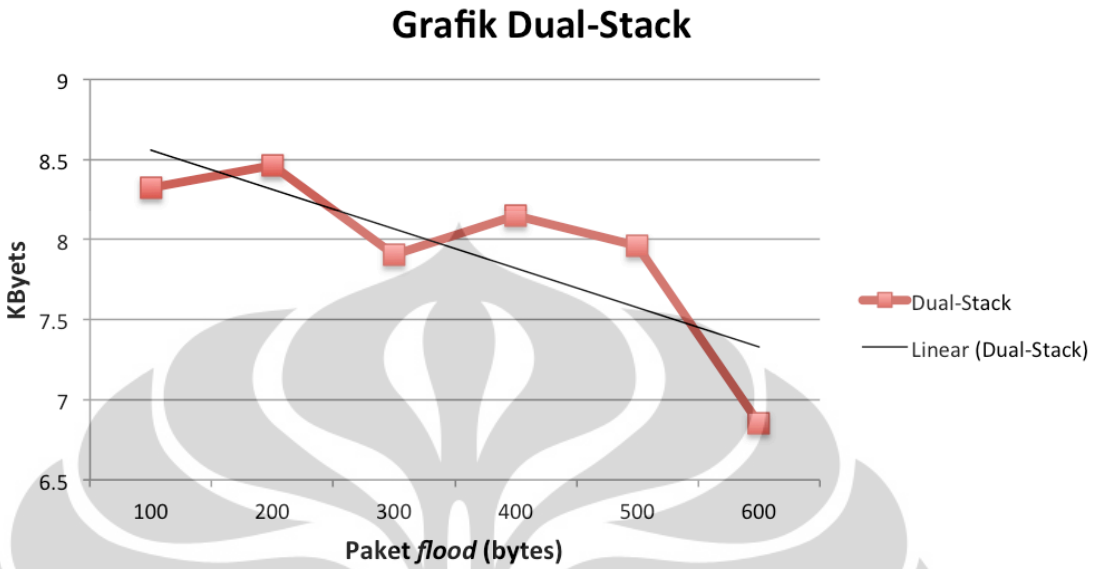


Gambar 4.5 Grafik *throughput* pada topologi GRE

Pada Gambar 4.5, nilai pada sumbu vertikal adalah nilai *throughput* dengan satuan Kbytes/sec sedangkan pada sumbu horizontal adalah besar paket *flood* dari serangan.

Gambar 4.5 menunjukkan adanya kecenderungan nilai *throughput* yang menurun pada saat melakukan transfer dokumen dari *client* FTP menuju Server. Penurunan nilai *throughput* pada jaringan yang menggunakan *dual-stack* lebih banyak dibandingkan dengan jaringan yang menggunakan *tunneling* GRE. Hal ini terlihat pada data terakhir pada topologi *dual-stack* pada Gambar 4.6. Untuk lebih jelasnya, perhitungan dilakukan dengan membuat *trend line* pada grafik GRE dan *dual-stack*. *Trend line* yang

digunakan pada grafik *throughput* adalah linear. Berikut adalah grafik *throughput* pada topologi *dual-stack* disertai *trend line* pada Gambar 4.6



Gambar 4.6 Grafik *throughput* pada topologi *dual-stack*

Pada Gambar 4.6 terlihat penurunan nilai *throughput* pada metode *dual-stack* yang menurun dengan kecenderungan linear. Terjadi penurunan nilai *throughput* pada jaringan yang menggunakan tunneling GRE sebesar 0.8013 Kbytes atau 9.27%. Untuk jaringan yang menggunakan metode *dual-stack*, terjadi penurunan nilai *throughput* 1.4674 Kbytes atau 17.62%.

4.4 Analisis Packet Loss

Parameter *packet loss* digunakan untuk mengetahui seberapa efektif paket dikirimkan sehingga perbandingan antara paket yang dikirim dan yang diterima bernilai kecil. *Packet loss* didapatkan dengan perhitungan berikut:

$$Packet\ Loss = \frac{Paket\ yang\ dikirim - Paket\ yang\ diterima}{Paket\ yang\ dikirim} \times 100\%$$

Hasil pengambilan data oleh wireshark untuk nilai *packet loss* sebagai berikut:

Tabel 4.2 Nilai *packet loss*

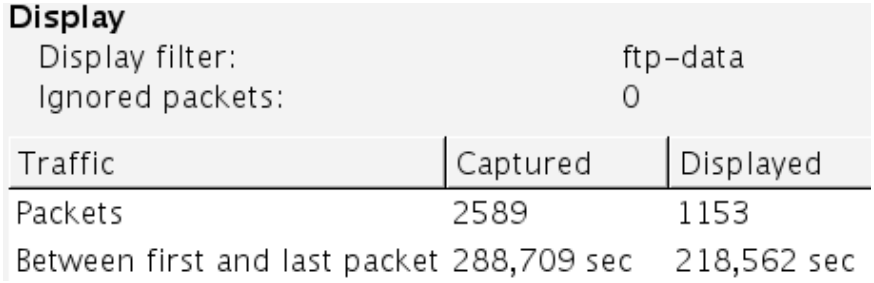
Packet Size	GRE	Dual-stack
100	12.0961%	7.0270%
200	10.6112%	6.5825%
300	7.1991%	4.1628%
400	4.9912%	7.0852%
500	7.9684%	8.2373%
600	8.4995%	7.3345%

Tabel 4.2 menunjukkan beragamnya nilai *packet loss* pada enam kali pengujian meskipun paket *flood* ditambah. Untuk mendapatkan nilai *packet loss*, dilakukan penjumlahan rata-rata pada masing-masing topologi. Perhitungan *packet loss* tidak dengan pengelompokkan data karena perubahan nilai pada masing-masing topologi tidak stabil.

Packet loss pada jaringan yang menggunakan *tunneling* FTP ketika dikenai serangan Denial of Service sebesar 8.56% sedangkan untuk jaringan yang menggunakan *dual-stack* sebesar 6.74%.

4.5 Analisis Delay

Delay adalah waktu yang dibutuhkan untuk pengiriman seluruh paket dibagi dengan total paket. Pada wireshark tidak semua paket yang termasuk untuk menentukan nilai *delay* melainkan dokumen yang dikirim saja dalam hal ini dokumen yang dikirimkan lewat aplikasi FTP. Parameter yang digunakan untuk menghitung delay adalah waktu tempuh total paket yang diperlihatkan pada wireshark di *toolbar* Summary dengan kunci 'between first and last packet'.



Gambar 4.7 Parameter untuk menghitung *delay*

Seperti terlihat pada Gambar 4.7, yang menjadi ukuran untuk penentuan *delay* adalah pada sisi kanan yaitu kolom 'Displayed'. Perhitungan nilai *delay* didapatkan dengan cara:

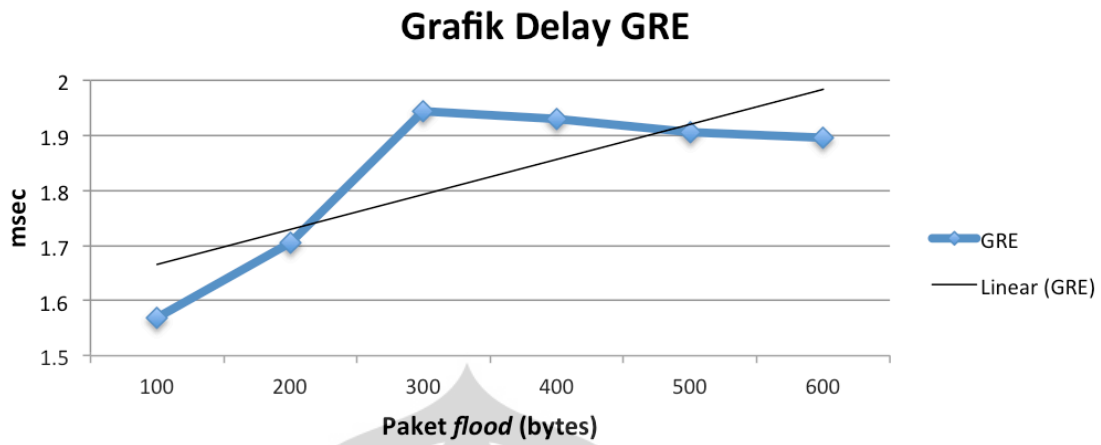
$$Delay = \frac{Transfer\ time}{Total\ Packet}\ (sec)$$

Transfer time pada Gambar 4.5 diperlihatkan pada baris 'Between first and last packet' sedangkan *Total* Packet adalah baris 'Paket'. Dari hasil perekaman aktifitas di wireshark didapatkan *delay* sebanyak enam data untuk masing-masing penggunaan metode transisi. Satuan yang digunakan adalah millisecond seperti terlihat pada Tabel 4.3.

Tabel 4.3 Nilai *delay*

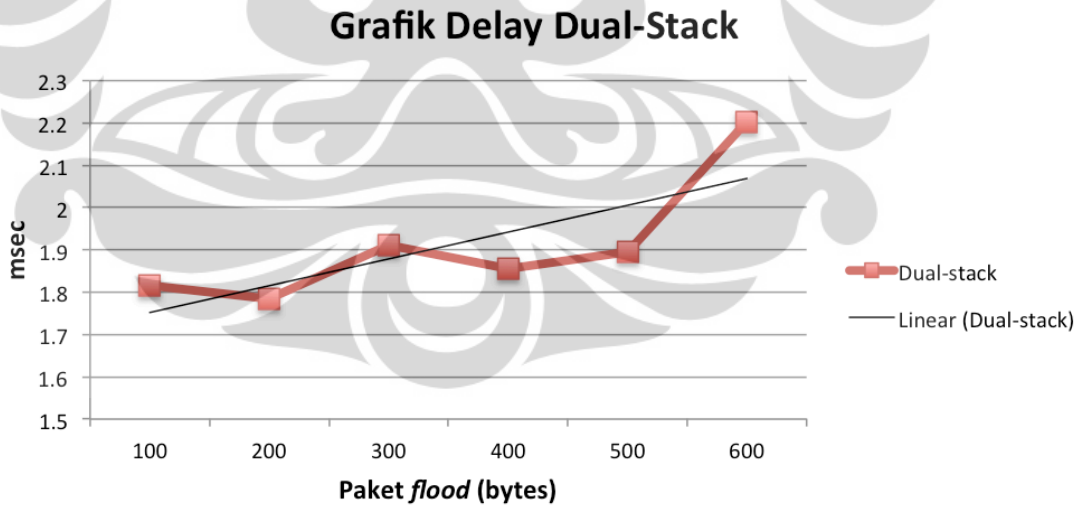
Packet Size (bytes)	GRE (msec)	Dual-stack (msec)
100	1.5688	1.8148
200	1.7045	1.7845
300	1.9431	1.9100
400	1.9300	1.8550
500	1.9064	1.8954
600	1.8955	2.2030

Parameter *delay* merupakan parameter yang signifikan diketahui perubahannya ketika terjadi serangan Denial of Service. Ketika suatu server diserang menggunakan paket *flood*, semakin besar ukuran paket semakin besar pula *delay* yang ditimbulkan. Hasil yang menunjukkan nilai *delay* ditampilkan pada Gambar 4.8 dibawah ini.



Gambar 4.8 Grafik kenaikan nilai *delay* pada topologi GRE

Grafik diatas menunjukkan nilai *delay* berangsur-angsur naik seiring bertambahnya ukuran paket *flood* yang menyerang Server. Jaringan yang menggunakan metode *dual-stack* berhenti sampai pengambilan data keenam karena penambahan ukuran paket *flood* membuat kegagalan pada transfer dokumen. Paket dari dokumen yang diterima server tidak utuh. Maka dari itu pengolahan dan analisis data dibatasi sampai serangan *flood* mencapai 600 bytes. Untuk melihat kecenderungan kenaikan *delay* pada dual-stack, dibuat *trend line* pada Gambar 4.9.



Gambar 4.9 Grafik *delay* pada topologi *dual-stack*

Dari hasil tersebut diketahui bahwa nilai *delay* naik untuk jaringan yang menggunakan *tunneling* GRE sebesar 0.3267 dan 0.3882 milisecond pada jaringan yang menggunakan *dual-stack*. Berdasarkan hasil perhitungan dan analisis, kenaikan nilai *delay* pada jaringan yang menggunakan *tunneling*

GRE sebesar 17.23% dan pada jaringan *dual-stack* mengalamai kenaikan sebesar 17.62%.

4.6 Analisis Keseluruhan Pengaruh Serangan Denial of Service

Setelah parameter-parameter pengukuran performa seperti *throughput, packet loss,* dan *delay* diketahui, serangan Denial of Service pada level tertentu berpengaruh signifikan terhadap turunnya kinerja jaringan IPv6. Pada nilai *throughput* diketahui bahwa penurunan nilai pada kedua topologi cukup berbeda bahkan mengalami kenaikan pada topologi GRE. Setelah data dikelompokkan diketahui pada rasio serangan 100 bytes – 600 bytes nilai *throughput* pada topologi GRE mengalami penurunan sebanyak 9.27%. Sedangkan pada *dual-stack* terjadi penurunan nilai *throughput* yang besar ketika paket *flood* bertambah.

Penurunan kemudian terjadi kenaikan pada topologi GRE disebabkan karena karakteristik GRE yang mampu membuat paket bertahan dalam trafik yang padat. Trafik yang padat disebabkan adanya *flooding* yang dilakukan PC Attacker pada jaringan AER. Selain itu karena cara kerja protokol GRE hampir mirip dengan cara kerja IPSec, bedanya adalah pada enkripsi.

Perubahan nilai *packet loss* pada setiap pertambahan paket *flood* cenderung tidak teratur untuk kedua topologi. Pertambahan serangan yang tidak terlalu besar tidak membuat pengaruh yang signifikan yang membuat nilai *packet loss* berangsur turun asalkan paket utuh sampai ke pada FTP Server. Jaringan yang terkena serangan Denial of Service memiliki nilai *packet loss* yang berbeda untuk setiap penggunaan mekanisme transisi dan nilai. Topologi *dual-stack* memiliki nilai *packet loss* yang lebih rendah dari topologi GRE.

Penurunan kinerja pada Jaringan IPv6 dengan parameter *delay* akibat *flooding* menaikkan nilai *delay* untuk masing-masing topologi. Pertambahan ukuran paket *flood* sebesar 100 bytes hanya menaikkan nilai *delay* yang tidak jauh berbeda untuk setiap data kecuali pada pengambilan data awal untuk topologi GRE. Perubahan nilai *delay* pada topologi *dual-stack* semakin tinggi pada pengambilan data terakhir. Disebabkan karena setelah serangan dinaikkan lebih dari 600 bytes dokumen gagal dikirim sehingga batas akhir ini membuat kerja protokol *dual-stack* meningkat.

Pengambilan data hanya dilakukan sampai serangan paket *flood* sebesar 600 bytes karena gagalnya pengiriman data dari *client* ke FTP Server ketika digunakan topologi *dual-stack*. Pengiriman gagal untuk paket *flood* lebih dari 600 bytes dengan sampel 650 bytes.

BAB V

KESIMPULAN

1. Perancangan ini dilakukan dengan menggunakan metode mekanisme transisi yang berbeda namun dengan *network environment* yang sama menggunakan IPv6. Percobaan dilakukan dengan dua kasus yaitu menggunakan *tunneling* GRE dan *dual-stack*.
2. Penurunan nilai *throughput* pada topologi GRE sebesar 9.27%. Sedangkan pada topologi *dual-stack* penurunan nilai *throughput* sebesar 17.62% dan lebih besar dari topologi GRE.
3. Pada parameter *packet loss* perbandingan antara topologi GRE dan *dual-stack* berbeda dari nilai *throughput*. Untuk topologi GRE nilai *packet loss* sebesar 8.36% dan pada topologi *dual-stack* nilai *packet loss* sebesar 6.74%. Nilai *packet loss* pada topologi *dual-stack* lebih rendah 24.03% dibanding penggunaan topologi GRE.
4. Dari pengukuran dan analisis parameter *delay*. Penurunan kinerja pada jaringan IPv6 ketika mendapat serangan *flooding* secara bertingkat untuk topologi GRE sebesar 0.3267 milisecond atau 17.23% dan untuk topologi *dual-stack* sebesar 0.3882 milisecond atau 17.62%. Besarnya delay pada topologi GRE jika dibandingkan dengan topologi *dual-stack* lebih kecil 15.84%
5. Dari hasil pengukuran dan analisis didapatkan bahwa pada mekanisme transisi penggunaan *tunneling* GRE lebih baik dibanding dengan penggunaan *dual-stack* mengacu pada penuranan nilai QoS yang dilakukan dengan uji coba transfer *file* pada aplikasi FTP.

DAFTAR REFERENSI

[1] Prawirayudha, Ardian. *Analisa Performa Jaringan Automatic 6to4 tunneling dan Jaringan Manually Configured IPv6 Tunneling dengan Menggunakan Aplikasi VLC dan Helix Streaming Server*. Universitas Indonesia. Skripsi. 2008: Depok

[2] Header pada IPv6. Diakses pada 8 November 2012. http://docs.oracle.com/cd/E19683-01/817-0573/chapter1-fig-8/index.html

[3] Bentuk tunneling secara umum. Diakses pada 14 November 2012. http://www.cisco.com/en/US/tech/tk872/technologies_configuration_example09186a00800b49a5.shtml

[4] ftp://ftp.hp.com/pub/networking/software/ProCurve-SR-dl-GRE-Config-Guide.pdf (Diakses pada 24 Desember 2012)

[5] ftp://ftp.hp.com/pub/networking/software/ProCurve-SR-dl-GRE-Config-Guide.pdf (Diakses pada 24 Desember 2012)

[6] Tunneling GRE. https://sites.google.com/site/amitsciscozone/home/ipsec/ipsec-with-ipv6-in-ipv4-gre-tunnel (Diakses pada 25 Desember 2012)

[7] Wan Nor, Ashikin Wan Ali. Distributed Security Policy for IPv6 Deployment. IEEE Paper. 2011: Malaysia

[8] http://www.docstoc.com/docs/56867475/What-is-Distributed-Denial-of-Service (Diakses pada 26 Desember 2012)

[9] http://adhit.web.id/cara-kerja-ftp-server-instalasi-dan-konfigurasi-vsftpd/ (Diakses pada 14 Januari 2013)

[10] A.H.M.Amin. *VoIP Performance Using QoS Parameters*. The Second International Conference on Innovations in Information Technology. 2005. Diakses pada tanggal: 14 November 2012. http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=A7AFA7F8843FECA082EC7910849D1E05?doi=10.1.1.118.6117&rep=rep1&type=pdf

[11] Meenakshi, S P., Raghavan, S V., Bhaskar S. M. A Study Path Behaviour Characteristics of IPv6 Based Reflector Attacks. IEEE: WNM Bonn. 2011

[12] http://citradayamaxima.com/?p=385 (Diakses pada 26 Desember 2012)

[13] http://cisco.com. (Diakses pada 26 Desember 2012)

[14] http://www.backtrack-linux.org/images/bt5-teaser01.png (Diakses pada 28 Desember 2012)

[15] http://www.petri.co.il/ipv6-header-vs-ipv4.htm (Diakses pada 1 Januari 2013)

[16] http://portableapps.com/apps/internet/filezilla_portable (Diakses pada 2 Januari 2013)

LAMPIRAN

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::2/64
!
```

```
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
```

DUAL-STACK

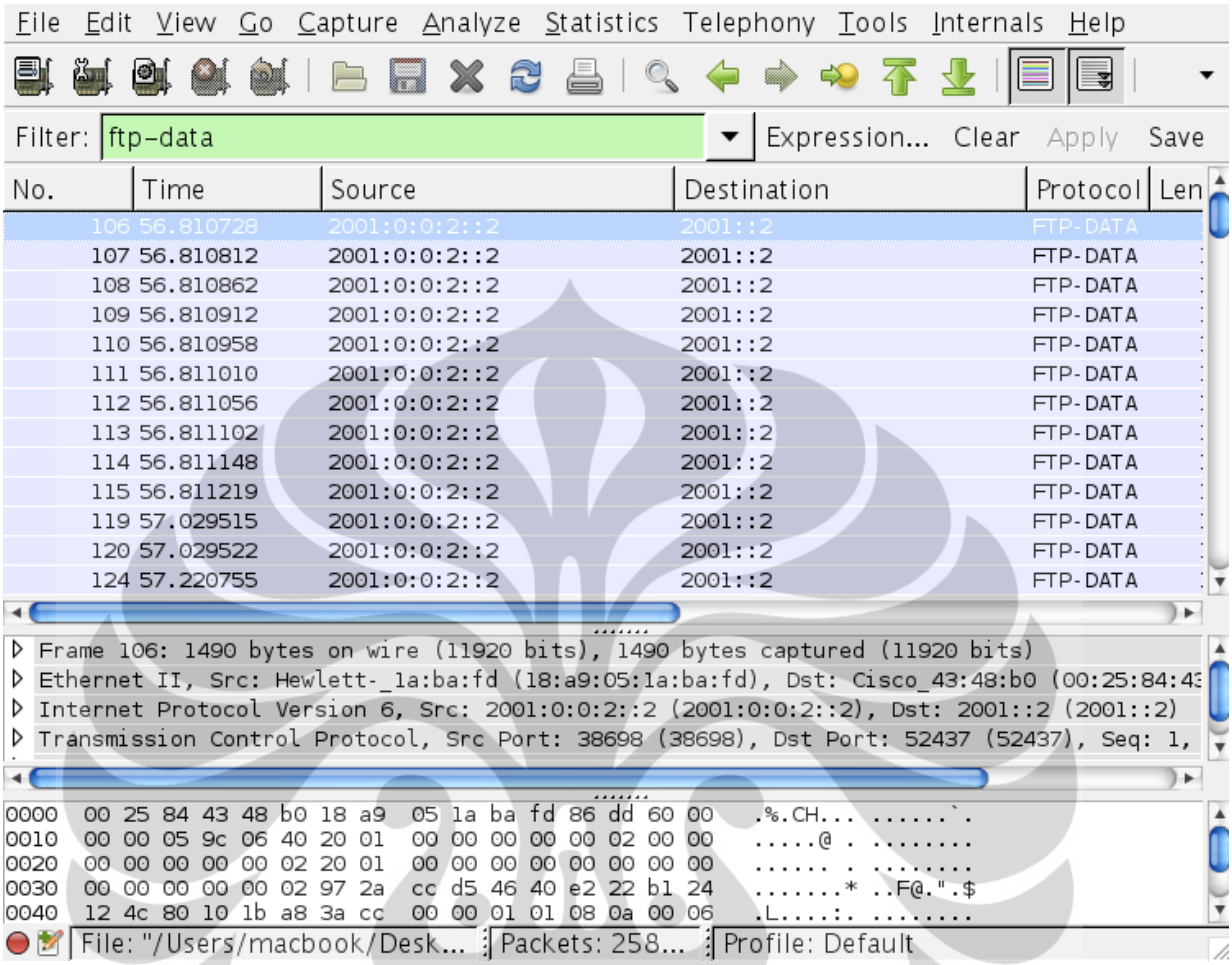
- Konfigurasi R1

```
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 FastEthernet0/0
 ip address 192.168.1.1 255.255.255.0
 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
 ipv6 address 2001:0:0:1::1/64
!
no ip http server
no ip http secure-server
!
!
ipv6 route 2001:0:0:2::/64 se0/0/1
ip route 192.168.3.0 255.255.255.0 se0/0/1
!
!
!
control-plane
!
!
!
!
!
!
gatekeeper
 shutdown
!
!
line con 0
 stopbits 1
line aux 0
line vty 0 4
!
!
end
```

Contoh *capture* Wireshark untuk Topologi GRE



Contoh *capture* Wireshark pada Topologi Dual-stack untuk serangan 100 bytes