UNIVERSITAS INDONESIA

ANALISA PERFORMANSI KEAMANAN JARINGAN *MOBILE* IPV6 MENGGUNAKAN *ROUTE OPTIMIZATION* DENGAN SERANGAN *DENIAL OF SERVICE* PADA APLIKASI FTP

SKRIPSI

MAHESA ADHITYA PUTRA
09006638396

FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK KOMPUTER
JUNI 2013

UNIVERSITAS INDONESIA

ANALISA PERFORMANSI KEAMANAN JARINGAN *MOBILE* IPV6 MENGGUNAKAN *ROUTE OPTIMIZATION* DENGAN SERANGAN *DENIAL OF SERVICE* PADA APLIKASI FTP

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

MAHESA ADHITYA PUTRA
0906638396

FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK KOMPUTER
JUNI 2013

HALAMAN PERNYATAAN ORISINALITAS

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

Nama	**: Mahesa Adhitya Putra**
NPM	**: 0906638396**
Tanda Tangan	**:** 
Tanggal	**: 9 Juni 2013**

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh	:	
Nama	:	Mahesa Adhitya Putra
NPM	:	0906638396
Program Studi	:	Teknik Komputer
Judul Skripsi	:	ANALISA PERFORMANSI KEAMANAN JARINGAN *MOBILE* IPV6 MENGGUNAKAN *ROUTE OPTIMIZATION* DENGAN SERANGAN *DENIAL OF SERVICE* PADA APLIKASI FTP

Telah dipresentasikan dan diterima sebagai bagian dari persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Komputer, Fakultas Teknik, Universitas Indonesia.

Pembimbing : Ir. A. Endang Sriningsih, M.T., Si. (.....)

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

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

Ditetapkan di : Depok

Tanggal : 14 Juni 2013

KATA PENGANTAR

Puji dan syukur saya ucapkan kepada Allah SWT yang berkat rahmat-Nya maka saya dapat menyelesaikan skripsi ini. Penulisan skripsi ini bertujuan untuk memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Program Studi Teknik Komputer pada Fakultas Teknik Universitas Indonesia. Saya ingin mengucapkan terima kasih kepada:

1) Ir. A. Endang Sriningsih, M.T., Si, selaku dosen pembimbing yang telah menyediakan waktu, tenaga dan pikiran sehingga saya dapat menyelesaikan penyusunan skripsi ini;
2) Papa, Mama dan keluarga saya yang telah memberikan bantuan baik secara doa, material dan moral;
3) Teman - teman seperbimbingan saya, Muhammad Iqbal, Ivan Farhan, Ihsan Nugraha, Aldiansah Prayogi dan Jihan Prama;
4) Teman - teman asisten lab saya, Alwin Muhaimin, Randi Kurnia Pranata, Armandya Rohadian Megantara dan Rizki Reynaldo Nasser;
5) Serta teman - teman saya yang telah banyak mendukung dan membantu selama proses penyelesaian skripsi ini.

Mohon maaf apabila terdapat kekurangan dan kesalahan dalam penulisan nama dan gelar serta kesalahan dalam penulisan skripsi ini. Karenanya, saya mengharapkan kritik dan saran yang dapat membangun skripsi ini. Semoga penulisan skripsi ini dapat membantu kontribusi bagi perkembangan ilmu pengetahuan dan teknologi di Indonesia.

Depok, 9 Juni 2013

Mahesa Adhitya Putra

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

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

Nama	:	Mahesa Adhitya Putra
NPM	:	0906638396
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:

ANALISA PERFORMANSI KEAMANAN JARINGAN *MOBILE* IPV6 MENGGUNAKAN *ROUTE OPTIMIZATION* DENGAN SERANGAN *DENIAL OF SERVICE* PADA APLIKASI FTP

Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalih media/format-kan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan skripsi saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenar-benarnya.

Dibuat di : Depok

Pada tanggal : 9 Juni 2013,

Yang menyatakan,

(Mahesa Adhitya Putra)

ABSTRAK

Nama	:	Mahesa Adhitya Putra
Program Studi	:	Teknik Komputer
Judul	:	ANALISA PERFORMANSI KEAMANAN JARINGAN *MOBILE* IPV6 MENGGUNAKAN *ROUTE OPTIMIZATION* DENGAN SERANGAN *DENIAL OF SERVICE* PADA APLIKASI FTP

Jaringan *Mobile Internet Protocol* adalah suatu fitur yang terdapat pada IPv6 yang memungkinkan *Mobile device* dapat diidentifikasikan menggunakan IP tunggal walaupun terjadi perpindahan koneksi dari satu jaringan (*Home Network*) ke jaringan lain (*Foreign Network)* tanpa mengganggu proses aplikasi yang sedang berjalan. Performa jaringan akan diuji menggunakan *threat* (serangan) dan diukur dengan parameter *transfer time, delay* dan *throughput.* Aplikasi yang digunakan adalah FTP (*File Transfer Protocol*). Hasil penyerangan tersebut dibandingkan antara sebelum dan sesudah dilakukan penyerangan di *Home Network* dan *Foreign Network*. Serangan yang paling besar pengaruhnya adalah di *Foreign Network* dengan *Denial of Service* 24 *thread*, dimana *transfer time* meningkat 31.05%, *delay* meningkat 25.38% dan *throughput* menurun 18.97% dibandingkan dengan sebelum diserang.

Kata kunci:

Mobile Internet Protocol, IPv6, FTP, Denial of Service, thread

ABSTRACT

Name	:	Mahesa Adhitya Putra
Study Program	:	Computer Engineering
Title	:	PERFORMANCE ANALYSIS OF NETWORK SECURITY IN MOBILE IPV6 DESIGN USING ROUTE OPTIMIZATION WITH DENIAL OF SERVICE ATTACK IN FTPAPPLICATION

Mobile Internet Protocol is a feature contained in IPv6 which enable Mobile *device* to be identified by using a single IP even though the connection is moved from a network (Home Network) to another network (Foreign Network) without intruding application processing. Network performance is tested using two *threat*s and measured using *transfer* time, delay and throughput as parameters. The application used in the form is FTP (*File Transfer* Protocol). The results will be compared before and after attacked in Home Network and Foreign Network. The result shows that the most affected network is Foreign Network with 24 *thread* of Denial of Service, where *transfer* time increased 31.05%, delay increased 25.38% and throughput decreased 18.97% compared to before attack.

Keyword:

Mobile Internet Protocol, IPV6, FTP, Denial of Service, *thread*

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB 1
PENDAHULUAN

1.1 Latar Belakang Masalah

Saat ini perkembangan teknologi sudah sangat berkembang pesat, terutama dalam teknologi komunikasi yang saat ini sedang marak dikembangkan oleh perusahaan-perusahaan yang bergerak di bidang ICT. Semakin *Mobile* suatu *device* yang digunakan, maka semakin praktis dan digemari oleh banyak orang. Dengan demikian seiring berjalannya perkembangan teknologi, maka hal ini tentunya mempengaruhi perkembangan IP.

IPv6 *(Internet Protocol* versi 6*)* merupakan sebuah protokol baru dalam penggunaan internet yang dikembangkan berdasarkan IPv4 (*Internet Protocol* versi 4*)*. Pengembangan ini didasarkan atas masalah terbatasnya jumlah alamat yang dimiliki oleh IPv4 yang hanya mempunyai panjang 32 bit. Sementara itu penggunaan internet terus bertambah dan permintaan akan IP terus bertambah sedangkan IP yang tersedia terbatas dan dapat dikatakan sudah habis. Berangkat dari hal inilah IPv6 dikembangkan dengan pengalamatan sepanjang 128 bit, yang memiliki ukuran 4 kali lipat lebih banyak daripada IPv4. Selain itu IPv6 memiliki pengalamatan yang lebih baik, sistem keamanan yang lebih baik, serta bisa digunakan pada aplikasi yang bersifat *real-time* bila dibandingkan dengan IPv4.

Internet Protocol memiliki kelebihan berupa *Mobile* IP, dimana *mobile* disini memiliki arti konektivitas pada *mobile device* yang dapat melintasi antar jaringan IP seperti pada *wireless*. Seiring berjalannya waktu dan berkembangnya aplikasi internet seperti VoIP *(Voice over* IP*), Video on Demand* dan aplikasi *real-time* lainnya, maka *Mobile* IP ini perlu dikembangkan pada IPv6. Salah satu aplikasi yang dapat diterapkan dalam IPv6 adalah FTP *(File Transfer Protocol)*.

File Transfer Protocol adalah suatu protokol pada jaringan yang berjalan di dalam lapisan aplikasi yang merupakan standar untuk pengiriman berkas *(file)*. FTP digunakan untuk melakukan pengunduhan *(download)* dan pengunggahan *(upload) file* antara klien FTP dan *server* FTP. Sebuah klien FTP dapat mengeluarkan

perintah-perintah FTP ke sebuah *server* FTP, sementara *server* FTP dengan sebuah *Windows Service* yang berjalan di atas sebuah komputer akan merespon perintah-perintah dari sebuah klien FTP. Perintah-perintah FTP dapat digunakan untuk menambah direktori, mengubah direktori, mengubah modus pengiriman antara biner dan ASCII, menghapus *file*, mengunggah berkas komputer ke *server* FTP, serta mengunduh berkas dari *server* FTP.

Walaupun dikatakan lebih baik tingkat keamanannya dibandingkan dengan IPv4, namun bukan berarti IPv6 ini aman sepenuhnya, tentunya juga terdapat celah pada sistem ini. Dengan menggunakan *threat* (serangan) yang ada seperti *Denial of Service* dapat diamati pengaruh performa pada sistemnya.

1.2 Tujuan

Tujuan penulisan skripsi ini adalah untuk mengukur dan menganalisa perfomansi pada jaringan *Mobile* IPv6 sederhana dengan metode *Route Optimization* lalu diimplementasikan dengan FTP *(File Transfer Protocol)* yang selanjutnya dilakukan penyerangan kepada jaringan dengan serangan *Denial of Service*.

1.3 Pembatasan Masalah

Batasan masalah dari skripsi ini adalah melihat performa jaringan pada *Mobile* IPv6 dengan metode *Route Optimization* menggunakan sistem operasi Linux Ubuntu dan berikut langkah-langkahnya:

1. Merancang sebuah jaringan dengan protokol *Mobile* IPv6 yang terdiri dari 3 PC, 2 laptop dan 2 *Access Point*.
2. Mengukur perfomansi aplikasi FTP (*File Transfer Protocol*) pada saat sebelum dilakukan dan sesudah dilakukan penyerangan menggunakan *Denial of Service*.
3. Melakukan pengukuran dan analisa parameter performansi jaringan IPv6, yaitu *transfer time, delay* dan *throughput*.

1.4 Metodologi Penelitian

Metode yang digunakan pada pembuatan skripsi ini adalah:

1. Studi Literatur

   Mempelajari buku, jurnal, paper, skripsi dan semua referensi lain yang berhubungan dengan IPv6 dan *Mobile* IPv6.

2. Perancangan Jaringan

   Perancangan jaringan *Mobile* IPv6 menggunakan *tools* yang ada dan membuat beberapa skenario yang mungkin terjadi sesuai penggunaan sesungguhnya, seperti melakukan gangguan - gangguan (interupsi) untuk mendapatkan data.

3. Pengukuran dan Analisa Data

   Mengukur dan menganalisa parameter QoS yang didapat dengan menggunakan perangkat lunak *Wireshark*.

4. Kesimpulan

   Mengambil kesimpulan dari hasil perbandingan serangan di *Home Network* dan *Foreign Network* dengan serangan *Denial of Service* pada FTP.

1.5 Sistematika Penulisan

Untuk memudahkan pembahasan, maka skripsi ini akan dibagi menjadi 5 bab dengan sistematika sebagai berikut:

- **BAB 1 : Pendahuluan**

  Bab ini menjelaskan Latar Belakang Masalah, Tujuan, Pembatasan Masalah, Metodologi Penulisan dan Sistematika Penulisan.

- **BAB 2 : Dasar Teori**

  Bab ini menjelaskan landasan teori mengenai IPv6, *Mobile* IPv6 serta hal-hal lain yang berkaitan secara lengkap.

- **BAB 3 : Perancangan Jaringan *Mobile* IPv6**

  Bab ini menjelaskan desain beserta peralatan dan komponen yang digunakan untuk menyusun jaringannya.

- **BAB 4 : Pengambilan dan Analisa Data**

  Pengambilan dan analisis data dengan parameter *transfer time*, *delay* dan *throughput* Jaringan *Mobile* IPv6

- **BAB 5 : Kesimpulan**

  Bab ini berisi kesimpulan yang dapat diambil dari skripsi ini.

BAB 2
MOBILE INTERNET PROTOCOL

2.1 Konsep Dasar Protokol Internet

Protokol dapat dikatakan sebagai suatu aturan yang digunakan untuk menyampaikan sesuatu dari satu pihak ke pihak lainnya. Dalam komunikasi, dapat diartikan sebagai usaha untuk melakukan kontak dari satu titik ke titik lainnya. Protokol berfungsi sebagai penerjemah antara keduanya sehingga pada akhirnya akan tercipta keterkaitan dan kesepahaman antara kedua belah pihak yang berkomunikasi.

Protokol Internet adalah protokol komunikasi utama yang digunakan untuk mengirimkan datagram (*network packets*) dengan melintasi jaringan internet dan bertanggung jawab untuk menentukan rute paket yang dikirim untuk sampai ke tujuan. Protokol Internet diperkenalkan pada tahun 1980-an yang digunakan untuk menghubungkan beberapa *Node* saja. Selanjutnya setelah disadari bahwa internet telah mendunia, akhirnya muncul berbagai macam protokol Internet.

2.2 IPv4 (*Internet Protocol* versi 4)

IPv4 yang merupakan singkatan dari *Internet Protocol* versi 4 adalah sebuah teknologi yang memungkinkan sebuah divais dapat terhubung ke internet. Ketika sebuah divais mengakses internet (PC, *Smartphone* atau divais lainnya), akan diberikan sebuah nomor atau identitas yang disebut dengan IP, misalnya 152.118.27.18. Untuk mengirim data dari satu divais ke divais lain melalui internet, sebuah data paket harus di*transfer* melewati jaringan yang mempunyai alamat IP. Tanpa alamat IP, divais tidak akan dapat berkomunikasi dan mengirim data ke divais lainnya. Alamat IP ini merupakan infrastruktur yang esensial dari jaringan.

IPv4 merupakan revisi keempat dalam pengembangan IP dan merupakan versi pertama dari protokol yang diluncurkan. IPv4 dideskripsikan dalam publikasi IETF (*Internet Engineering Task Force*) RFC 791 (September 1981), menggantikan definisi sebelumnya (RFC 760, Januari 1980). IPv4 merupakan sebuah protokol yang digunakan untuk Ethernet. IP beroperasi dengan model *best effort* (usaha terbaik),

yang berarti tidak menjamin pengiriman data. Aspek ini termasuk integritas data, dialamatkan oleh layer yang lebih atas, yaitu layer trans*port* seperti TCP (*Transmission Control Protocol*).

IPv4 menggunakan 32 bit untuk pengalamatannya, artinya IPv4 dapat mendukung sebanyak 2^{32} alamat IP dengan total sekitar 4,29 miliar pengalamatan (4.294.967.296). Jumlah tersebut sekilas terlihat banyak, tapi sekitar 4,29 miliar alamat tersebut telah dialokasikan ke berbagai institusi. Dengan perkembangan teknologi yang sangat pesat dan kebutuhan dalam penggunaan internet, saat ini kesemua alamat IP tersebut sudah hampir habis.

2.3 IPv6 (*Internet Protocol* versi 6)

IPv6 (*Internet Protocol* versi 6) adalah protokol internet versi yang lebih baru daripada IPv4 (*Internet Protocol* versi 4). IPv6 ini didesain untuk menggantikan IPv4. IPv4 memiliki kapasitas alamat sebesar 32 bit, sedangkan IPv6 memiliki kapasitas sebesar 128 bit, 4 kali lipat lebih banyak. Dengan demikian maka akan didapat jauh lebih banyak kapasitas pengalamatan yang memungkinkan internet akan terus berkembang. Pemilihan rute pengiriman bisa dilakukan secara efisien karena IPv6 memiliki tipe alamat *Anycast*.

IPv6 dapat secara otomatis melakukan berbagai *Setting* secara standar maupun *default*. Secara otomatis terdapat dua cara berdasarkan penggunaan alamat yaitu *Stateless* dan *Stateful*.

1. *Setting Stateless*

   Pada cara ini yang diperlukan adalah mengatur *router* dimana *host* yang telah tersambung di jaringan memperoleh prefiks dari alamat jaringan tersebut. *Server* tidak perlu disediakan untuk mengelola dan membagi IP *address*. Kelebihan *Setting Stateless* adalah lebih mudah untuk mengelola Ethernet atau FDDI karena hanya perlu minimal 48 bit atau setara dengan besar MAC *address*. Kelemahannya adalah tidak efisien dalam penggunaan alamat.

2. *Setting Stateful*

Pada cara ini yang diperlukan adalah menyediakan *server* untuk mengelola *range* IP *address* yang diberikan kepada *host* secara ketat. Informasi yang dibutuhkan adalah ICMP (*Internet Control Message Protocol*) yang telah diperluas antara *router, server* dan *host*.

Alamat pada IPv6 dibagi menjadi 3, yaitu *Unicast address, Multicast address* dan *Anycast address*. *Unicast address* adalah pengalamatan untuk komunikasi satu ke satu (*point-to-point*) dengan menunjuk salah satu dari *host* tersebut. *Multicast address* adalah pengalamatan untuk komunikasi satu ke banyak (*one-to-many*) dengan menunjuk salah satu *host* di grup tersebut. *Anycast address* adalah pengalamatan untuk komunikasi penyampaian paket data kepada anggota terdekat dari sebuah grup (*one-to-one-of-many*).

1. *Unicast Address*

Unicast address menyediakan komunikasi dalam jaringan secara langsung antar 2 *host*. *Unicast address* sendiri dibagi menjadi 3, yaitu:

a. *Link Local* yaitu jenis alamat yang mengizinkan komputer dapat berkomunikasi dengan komputer lainnya dalam satu *subnet*.
b. *Site Local* yaitu jenis alamat yang mengizinkan komputer dapat berkomunikasi dengan komputer lainnya dalam sebuah intranet.
c. *Global Address* yaitu jenis alamat yang mengizinkan komputer dapat berkomunikasi dengan komputer lainnya dalam internet berbasis IPv6.

2. *Multicast Address*

Multicast address menyediakan komunikasi dalam jaringan dari satu *host* ke banyak *host* dalam satu grup yang sama. *Multicast address* digunakan pada kebutuhan komunikasi one to many.

3. *Anycast Address*

Anycast address menyediakan komunikasi dalam jaringan dari satu *host* ke anggota *host* terdekat dari grup. *Anycast address* digunakan pada kebutuhan komunikasi *one to many*, bedanya dengan *Multicast address* adalah alamat tujuan hanya dikirim ke *router*, tidak langsung ke *host-host*.

Pengalamatan IPv6 berbeda dengan IPv4, hal ini dikarenakan model yang digunakan adalah x:x:x:x:x:x:x:x dimana setiap "x" tersebut memiliki nilai heksadesimal dari 16 bit, maka totalnya adalah 128 bit dan direpresentasikan dengan 8 bagian. Contoh pengalamatan pada IPv6 adalah:

2001 : db8 : ffff : 100a : 0000 : 0000 : 0000 : 0000

Dari contoh alamat di atas, terdapat 2 bagian yang terdiri dari 4 buah angka 0, maka penulisan alamat tersebut dapat disederhanakan menjadi 1 buah angka 0 saja, sehingga pengalamatannya akan menjadi:

2001 : db8 : ffff : 100a : 0 : 0 : 0 : 0

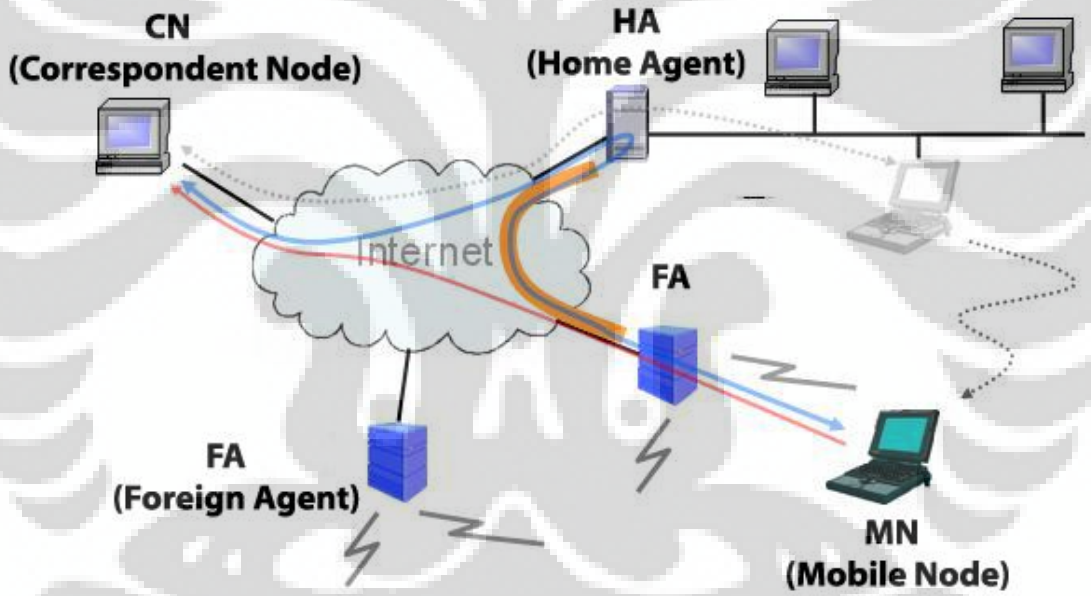
Bit awal pada alamat IPv6 membentuk prefiks *network*, tetapi berbeda dengan alamat pada IPv4, pada IPv6 tidak merujuk kepada *subnet mask* karena memang tidak mendukung *subnet mask*. Prefiks merupakan bagian dari alamat IP, di mana bit-bit tersebut mempunyai nilai yang tetap karena merupakan bagian dari sebuah rute atau *subnet identifier*. Prefiks dalam IPv6 direpresentasikan 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:

2001 : db8 : ffff : 100a : : /64

Berdasarkan contoh tersebut, maka 64 bit pertama dari alamat IPv6 tersebut merupakan prefiks alamat dan 64 bit sisanya merupakan *interface* ID. Dari alamat tersebut maka alamat tersebut memiliki *range* alamat dari 2001 : db8 : ffff : 100a :: sampai 2001 : db8 : ffff : 100a : ffff : ffff : ffff : ffff.

2.4 *Mobile Internet Protocol*

Mobile Internet Protocol adalah protokol standar komunikasi yang dirancang oleh IETF (*Internet Engineering Task Force*) yang mengizinkan pengguna *Mobile device* untuk pindah dari satu jaringan (*Home Network*) ke jaringan lain (*Foreign Network*) dengan mempertahankan IP *address*. Fitur ini terdapat baik pada IPv4 maupun IPv6. Setiap *Mobile Node* diidentifikasi oleh *Home address* tanpa mempedulikan lokasi di internet. Selama tidak berada di *Home Network*, sebuah *Mobile Node* dihubungkan dengan *Correspondent Node* dengan menggunakan sebuah *Care of Address* yang mengidentifikasikan lokasi sekarang, sementara *Home address* dihubungkan dengan *end point local* dari sebuah jalur ke *Home Agent*. Gambar 2.1 mengilustrasikan *Mobile Internet Protocol*:



Gambar 2.1 *Mobile Internet Protocol* [1]

2.4.1 Komponen *Mobile Internet Protocol*

Komponen yang dibutuhkan oleh sebuah *Mobile* IP adalah:

1. *Home Address*

   Home Address merupakan IP *address* yang diberikan pada *Mobile Node* ketika berada di *Home Network* maupun *Foreign Network*.

2. *Home Agent* (HA)

   Home Agent merupakan sebuah *Router* pada *Home Network* yang menentukan jalur informasi untuk mengirim ke *Mobile Node* ketika pindah dari *Home Network* dan mempertahankan informasi untuk *Mobile Node*. *Home Agent* biasa digunakan dengan satu atau lebih *Foreign Agent*.

3. *Home Network* (HN)

   Home Network merupakan sebuah jaringan dimana sebuah *Mobile Node* mendapatkan IP *Address*-nya. *Home Network* merupakan sub-*network* dari *Home Agent*. *Home Address* ditentukan menggunakan *Home Subnet Prefix*.

4. *Foreign Agent* (FA)

   Foreign Agent merupakan sebuah *Router* yang menyimpan informasi *Mobile Node* yang melewati jaringannya. *Foreign Agent* juga memperlihatkan *Care of Address* yang digunakan oleh *Mobile* IP.

5. *Foreign Network* (FN)

   Foreign Network merupakan sebuah jaringan dimana sebuah *Mobile Node* yang beroperasi sedang berada di luar *Home Network*.

6. *Mobile Node* (MN)

   Mobile Node merupakan *Node* IPv6 yang dapat berpindah konektivitas dan memiliki informasi *Home Address* dan alamat global pada lokasi dimana *Mobile Node* itu berada sekarang. *Mobile Node* mengidentifikasi

informasi pemetaan *Home Address* atau alamat sekarang dari *Home Agent* dan *Node* IPv6 lainnya yang sedang berkomunikasi dengan *Mobile Node*.

7. *Correspondent Node* (CN)

Correspondent Node merupakan sebuah *Node* yang dapat berkomunikasi dengan *Mobile Node* ketika berada di *Home Network* maupun di *Foreign Network*. *Correspondent Node* juga dapat berfungsi sebagai *Mobile Node* ataupun *Server*.

8. *Care of Address* (CoA)

Care of Address merupakan alamat yang digunakan pada saat *Mobile Node* terhubung dengan *Foreign Network*. *Care of Address* adalah kombinasi dari *Foreign Subnet Prefix* dan *interface* ID yang dimiliki oleh *Mobile Node*.

9. *Agent Advertisement* (AA)

Agent Advertisement merupakan sebuah pemberitahuan yang berisi informasi bagi *Mobile Node* untuk dapat terhubung ke *Mobile Agent.*

2.4.2 Cara Kerja *Mobile Internet Protocol*

Mobile IP bekerja dengan cara mengizinkan suatu *Mobile Node* untuk dapat memiliki dua IP *address*, yang pertama IP *address* tetap (*Home Address*) dan IP *Care of Address* (alamat sementara) yang digunakan untuk menghubungi jaringan dimana *Mobile Node* sedang berada. *Home Agent* menyimpan secara permanen informasi mengenai *Mobile Node* sementara *Foreign Agent* memberitahukan dan menyimpan alamat sementara *Mobile Node* yang mengunjungi jaringannya. [4]

Home Network terkait dengan *Home Agent* dimana *frame* akan selalu diteruskan ke *Home Agent* menggunakan mekanisme routing IP. Ketika *Home Agent* menerima paket ini, maka paket akan dialihkan terhadap *Foreign Agent* dengan

mengambil IP *Care of Address* dari tabel *look-up*. Selanjutnya paket IP yang asli akan ditambahkan header IP oleh *Home Agent*.

2.4.3 Kelebihan dan Kekurangan *Mobile* IP

Kelebihan *Mobile* IP diantaranya adalah bersifat *Mobile*, maka pengguna dapat menggunakan *device wireless* tersebut dimana saja. *Mobile* IP memudahkan pengguna mengakses internet di tempat yang jauh dan dapat memakai *router setup* maupun modem. Keuntungan lainnya adalah alamat *Mobile* IP akan dinamis, yang memungkinkan pengguna dapat terhubung ke internet tanpa memerlukan IP yang statis.

Adapun kekurangan pada *Mobile* IP yaitu ketersediaan internet tidak akan merata, karena biasanya akses tersebut berada di area kota besar. Kota kecil belum tentu tersedia akses yang dibutuhkan sehingga internet akan sulit untuk dijangkau.

2.4.4 Pengiriman Paket *Mobile* IPv6

Ketika *Mobile Node* tidak terhubung secara fisik ke *Home Network*, *Mobile Node* dapat menerima data dari dan mengirim data ke *Correspondent Node* menggunakan *Bidirectional Tunneling* dan *Route Optimization*.

1. *Bidirectional Tunneling*

   Bidirectional Tunneling digunakan pada dua kondisi, ketika *Correspondent Node* tidak memiliki *binding* untuk *Mobile Node* (proses registrasi sedang berlangsung) atau ketika *Correspondent Node* tidak mendukung *Mobile* IPv6. Dengan demikian *Bidirectional Tunneling* memastikan bahwa *Mobile Node* selalu terjangkau ketika tidak berada di *Home Network*, bahkan ketika *Correspondent Node* tidak memadai untuk *Mobile* IPv6. Pada metode ini *Correspondent Node* mengirim paket data ke *Home Address* dari *Mobile Node* tersebut. *Home Agent* menerima dan mengarahkan paket menggunakan jalur IPv6 ke *Care of Address Mobile Node*.

2. *Route Optimization*

Route Optimization adalah metode yang digunakan setelah proses registrasi *Correspondent (binding update)* selesai, maka *Mobile Node* dan *Correspondent Node* akan dapat berkomunikasi satu sama lain secara langsung tanpa harus mengirim data terlebih dahulu melalui *Home Agent* dan pengarahan jalur antara *Home Agent* dan *Mobile Node* tidak akan diperlukan. Ketika metode ini digunakan, tidak ada penambahan *latency* karena paket telah ditentukan jalurnya dari *Mobile Node* sebagai paket IP biasa. *Route Optimization* membutuhkan bantuan *Correspondent Node* yang mendukung protokol *Mobile* IPv6.

Pertama-tama paket data akan dikirim dari *Mobile Node* ke *Correspondent Node*. Dalam kasus ini, *Mobile Node* sudah melakukan *binding update*. Paket tersebut berisi:

- Header IPv6, berisi alamat sumber yang telah diatur untuk *Care of Address* dari *Mobile Node* dan alamat tujuannya adalah alamat *Correspondent Node*.
- *Destination Options Header*, berisi *Home option address* yang berisi *Home Address* dari *Mobile Node*. Dengan menggunakan *Home* Option *Address* maka alamat tujuan paket akan digantikan oleh *Home Address* dari *Mobile Node*.
- *Upper layer* PDU, berisi aplikasi layer data yang dikirim dari *Mobile Node* ke *Correspondent Node*. Pada layer aplikasi, terlihat seperti data dialamatkan dari *Home Address* dari *Mobile Node* ke *Correspondent Node*.

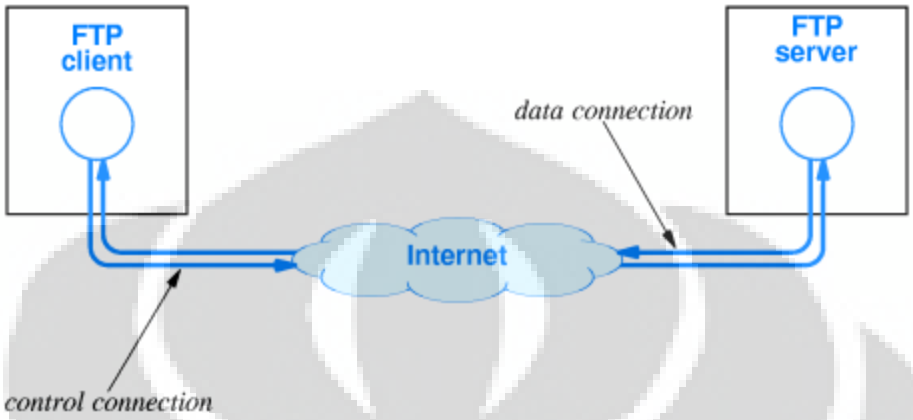
Selanjutnya paket data akan dikirim dari *Correspondent Node* ke *Mobile Node*. Lalu diasumsikan bahwa *Correspondent Node* sudah melakukan *binding cache entry* untuk *Home Address* dari *Mobile Node*, sehingga paket data tersebut berisi:

- Header IPv6, berisi alamat sumber dari alamat *Correspondent Node* dan alamat tujuannya adalah *Care of Address* dari *Mobile Node*. Karena *Care of Address* yang digunakan dan bukan *Home Address*, maka paket akan langsung dikirim langsung ke alamat *Mobile Node* sekarang.
- *Routing header* tipe 2, *Home Address* diatur untuk *Home Address* milik *Mobile Node*. Ketika menerima paket, *Mobile Node* membuang routing header tipe 2 dan menggantinya dengan *Care of Address* dengan *Home Address* sebagai tujuan dari header IPv6.
- Upper Layer PDU, berisi aplikasi layer data yang dikirim dari *Correspondent Node* ke *Mobile Node*. Pada layer aplikasi, terlihat bahwa data dialamatkan dari *Correspondent Node* ke *Home Address* milik *Mobile Node*.

2.5 FTP (*File Transfer Protocol*)

FTP (*File Transfer Protocol*) adalah protokol yang digunakan untuk memindahkan *file* diantara dua *host* di TCP/IP. Diantara kedua *host* tersebut dibuat sesi terlebih dahulu sebelum *transfer* data dimulai, barulah komunikasi dapat dilakukan antara *server* dan klien menggunakan TCP. FTP menggunakan Telnet protocol dalam mengontrol koneksinya. [5]

Port TCP yang biasa digunakan adalah *port* 20 dan 21. *Port* 21 memiliki beberapa fungsi, yaitu digunakan sebagai control *port* untuk membuat sebuah koneksi klien-*server*, mengizinkan klien untuk mengirimkan sebuah perintah FTP ke *server*, dan fungsi lainnya adalah mengembalikan respon *server* ke perintah tersebut. *Port* 20 digunakan untuk membentuk suatu koneksi baru dengan klien untuk men*transfer* data aktual yang sedang dipertukarkan saat melakukan pengunduhan (download) dan pengunggahan (upload). Gambar 2.2 mengilustrasikan hubungan antara klien FTP dan *server* FTP:



Gambar 2.2 Hubungan antara Klien FTP dan *Server* FTP [2]

Koneksi pada FTP ditetapkan dalam dua mode berbeda, yaitu modus aktif (*PORT*) dan modus pasif (PASV).

1. Modus aktif

Server FTP diatur untuk menggunakan sambungan di mode aktif dengan menunggu klien FTP membuka *port* dinamis. Lalu klien mengirimkan sebuah perintah *PORT* berisi *port* number dinamis yang berisi *control stream* dan menunggu sambungan dari *server* FTP. Ketika *server* FTP memulai sambungan data ke klien FTP mengikat *port* sumber ke *port* 20 di *server* FTP.

Dari sudut pandang *firewall server-side*, ada beberapa *port* yang harus dibuka untuk mendukung modus aktif FTP:

- *Port* 21 *server* FTP dari mana saja (klien memulai koneksi)
- *Port* 21 *server* FTP ke *port* 1023 (*server* bereaksi terhadap *port* kontrol klien)
- *Port* 20 *server* FTP ke *port* 1023 (*server* memulai koneksi data ke *port* data klien)

- *Port* 20 *server* FTP dari *port* 1023 (klien mengirim data ke *port* data *server*)

2. Modus Pasif

Saat suatu koneksi dibuka pada mode pasif, FTP *server* tidak menunggu FTP klien untuk mengirim informasi *transfer* data. *Server* menggunakan perintah PASV dan mengirimkan IP *address server* ke klien FTP untuk menghubungkan kemana dan dimana *port* tersebut digunakan. Dalam hal ini, FTP mengikat sambungan *port* sumber ke *port* dinamis. Modus PASV dirancang untuk klien FTP di belakang firewall.

Dari sudut pandang *firewall server-side*, untuk mendukung modus pasif FTP saluran komunikasi yang perlu dibuka adalah sebagai berikut:

- *Port* 21 *server* FTP dari mana saja (Klien memulai koneksi)
- *Port* 21 *server* FTP ke *port* 1023 (*Server* bereaksi terhadap *port* kontrol klien)
- *Port* 1023 *server* FTP dari mana saja (Klien memulai koneksi data ke *port* acak yang ditentukan oleh *server*).
- *Port* 1023 *server* FTP ke remote *port* (*Server* mengirimkan data ke *port* data klien)

Akan tetapi, beberapa administrator jaringan menonaktifkan mode PASV *server* FTP karena risiko keamanan.

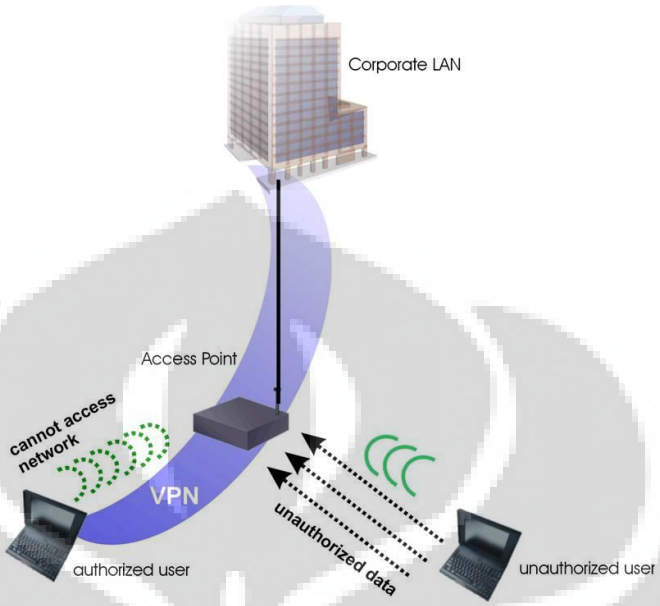
FTP sebenarnya cara yang tidak aman untuk men*transfer file* karena *file* tersebut di*transfer* tanpa melalui enkripsi terlebih dahulu tetapi melalui *clear text*. Mode text yang dipakai untuk *transfer* data adalah format ASCII atau format biner. Secara *default*, FTP menggunakan mode ASCII untuk *transfer* data. Karena pengirimannya tanpa enkripsi, maka *username, password*, data yang di*transfer*, maupun perintah yang dikirim dapat di *sniffing* oleh orang dengan menggunakan *protocol analyzer* (*Sniffer*). Solusi yang digunakan adalah dengan menggunakan SFTP (SSH FTP) yaitu FTP yang berbasis pada SSH atau menggunakan FTPS

(FTP *over* SSL) sehingga data yang dikirim terlebih dahulu dienkripsi (dikodekan).

2.6 *Security* pada *Mobile* IP

Security pada *Mobile* IP penting untuk diperhatikan, karena dengan sistem yang *Mobile* tentunya akan sulit untuk menjaga keamanan dan kerahasiaan data yang dikirim. Serangan biasanya ditujukan ke layer *network* ataupun data link. Pada skripsi ini akan dibahas serangan *Denial of Service* ke *Mobile* IPv6.

DoS (*Denial of Service*) adalah serangan yang digunakan untuk memonopoli *network resources*. Serangan ini membuat pengguna yang memiliki otorisasi tidak dapat mengakses jaringan yang dikehendaki. Penyerang juga dapat mengirimkan *binding update* palsu dengan tujuan menaikan trafik yang tidak diinginkan pada jaringan tersebut. Penyerang mencari sebuah situs dengan data stream yang berat dan menghubungkan ke jaringan tersebut. Selanjutnya penyerang mengirim *binding update* ke *Correspondent Node* untuk mengalihkan *arbitrary Node* lokasi baru penyerang. *Arbitary Node* ini kemudian akan diserang dengan banyak paket yang tidak penting. Penyerang juga dapat melakukan serangan sehingga jaringan akan mengakses data yang bukan dituju oleh pengguna yang mengakses. Gambar 2.3 mengilustrasikan serangan *Denial of Service*:

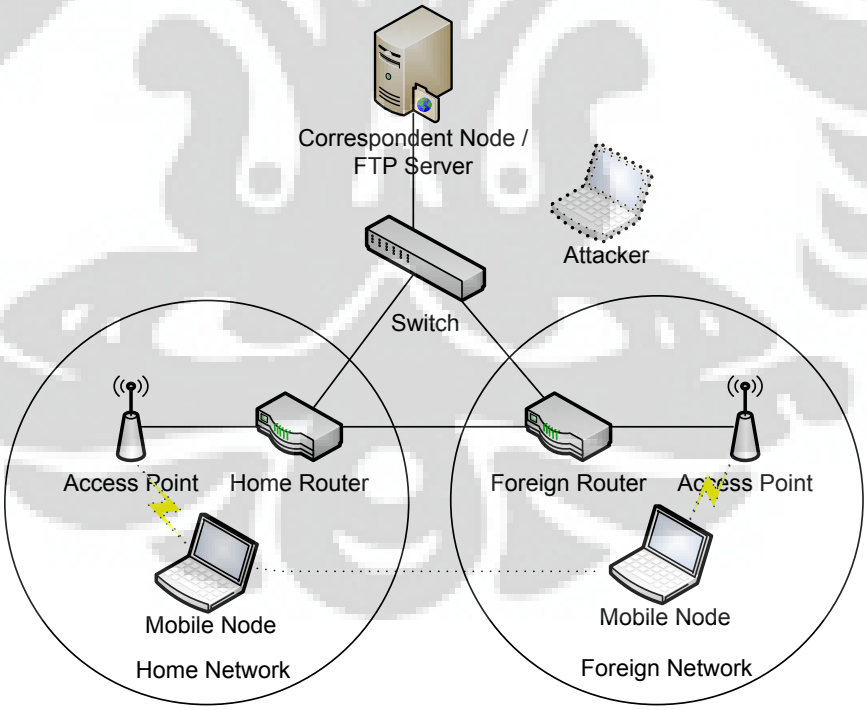


Gambar 2.3 Serangan DoS [3]

BAB 3
PERANCANGAN DAN IMPLEMENTASI JARINGAN *MOBILE* IPV6 UNTUK FTP (*FILE TRANSFER PROTOCOL*)

3.1 Topologi Jaringan

Topologi jaringan yang akan diimplementasikan adalah jaringan MIPv6 dengan menerapkan 2 macam serangan dengan metode *Route Optimization*. Topologi ini terdiri dari 3 PC, 2 laptop dan 2 *Access Point*. Masing-masing dari perangkat tersebut memiliki fungsi yang berbeda. 1 laptop sebagai *Mobile Node*, 1 laptop berfungsi sebagai *Attacker* yang melakukan *threat* kepada *Correspondent Node*, 1 PC berfungsi sebagai *Correspondent Node* atau FTP *Server*, 2 PC berfungsi sebagai *router* yang menghubungkan semua *device* yang bersangkutan, dimana 1 PC sebagai *Home Agent* dan 1 PC sebagai *Foreign Agent*, 1 *Access Point* berperan di *Home Network* dan 1 *Access Point* berperan di *Foreign Network*. Gambar 3.1 berikut mengilustrasikan rencana topologi jaringannya:



Gambar 3.1 Topologi Jaringan

- Skenario pertama

  Pada skenario pertama, jaringan akan dikonfigurasikan dengan *Mobile* IPv6 dan menggunakan metode *Route Optimization*, lalu dilakukan pengukuran performansi FTP pada jaringan ketika *Mobile Node* berada di *Home Network* sebelum dilakukan serangan.

- Skenario kedua

  Pada skenario kedua, jaringan akan dikonfigurasikan dengan *Mobile* IPv6 dan menggunakan metode *Route Optimization*, lalu dilakukan pengukuran performansi FTP pada jaringan ketika *Mobile Node* berada di *Foreign Network* sebelum dilakukan serangan.

- Skenario ketiga

  Pada skenario ketiga, jaringan akan dikonfigurasikan dengan *Mobile* IPv6 dan menggunakan metode *Route Optimization*, lalu jaringan akan diserang dengan *threat Denial of Service*, selanjutnya dilakukan pengukuran performansi FTP ketika *Mobile Node* berada di *Home Network*.

- Skenario keempat

  Pada skenario keempat, jaringan akan dikonfigurasikan dengan *Mobile* IPv6 dan menggunakan metode *Route Optimization*, lalu jaringan akan diserang dengan *threat Denial of Service*, selanjutnya dilakukan pengukuran performansi FTP ketika *Mobile Node* berada di *Foreign Network*.

3.2 Spesifikasi Sistem

3.2.1 Spesifikasi Hardware

Hardware yang akan digunakan dalam perancangan jaringan *Mobile* IPv6 ini adalah:

1. *Correspondent Node* / FTP *Server*

   Correspondent Node menggunakan sebuah PC sebagai FTP *Server*.

   Prosessor : Intel® Dual-Core™ CPU E6300 @2.80 GHz

   Memori : 2 GB

   Harddisk : 320 GB

2. *Home Router*

   Router yang digunakan adalah sebuah PC yang difungsikan sebagai PC *Router* yang mendukung *Mobile* IPv6 dan berfungsi sebagai penghubung antar semua *device* yang bersangkutan. PC ini digunakan sebagai *Home Router* dan *Home Agent*.

   Prosessor : Intel® Dual-Core™ CPU E6300 @2.80 GHz
   Memori : 2 GB
   Harddisk : 320 GB

3. *Foreign Router*

   Router yang digunakan adalah sebuah PC yang difungsikan sebagai PC *Router* yang mendukung *Mobile* IPv6 dan berfungsi sebagai penghubung antar semua *device* yang bersangkutan. PC ini digunakan sebagai *Foreign Router* dan *Foreign Agent*.

   Prosessor : Intel® Dual-Core™ CPU E6300 @2.80 GHz
   Memori : 2 GB
   Harddisk : 320 GB

4. *Access Point*

   Access Point yang digunakan tidaklah harus spesifik jenis tertentu dan digunakan untuk menghubungkan *Mobile Node* dengan *router* di *Home Network* dan *Foreign Network.*

   Tipe : TP-Link *Wireless*-N *Access Point* [TL-WA730RE]
   Data Rates : 54 Mbps

5. *Switch*

   Switch digunakan untuk menghubungkan *Home Agent*, *Foreign Agent* dan *Correspondent Node* / FTP *Server*.

   Tipe: TP-LINK-TL-SF1005D

   Port: 5-*port*s/10/100/Mbps

6. *Mobile Node*

   Mobile Node menggunakan laptop yang mendukung teknologi *wireless*.

   Prosessor : Intel® Core 2 Duo

   RAM : 2 GB

   Harddisk : 500 GB

   Wireless : Atheros *Wireless Network*

7. *Attacker*

   Attacker menggunakan sebuah laptop yang berfungsi untuk melakukan serangan terhadap jaringan dan mendukung teknologi *wireless*.

   Prosessor : Intel (R) Core i5

   RAM : 4GB

   Harddisk : 750 GB

   Wireless : Atheros *Wireless Network*

3.2.2 Spesifikasi Software

Software yang akan digunakan untuk perancangan jaringan *Mobile* IPv6 adalah:

1. Sistem Operasi *Linux Ubuntu* 13.04 LTS (*Raring Ringtail*)

   Sistem operasi Linux ini akan digunakan untuk mendukung jaringan *Mobile* IPv6. Alasan penggunaan sistem operasi ini adalah mudah untuk dikonfigurasikan karena kernel Linux dapat digunakan untuk berbagai macam modul konfigurasi.

2. UMIP (Linux *Mobile* IPv6 *Daemon*)

   UMIP yang dikenal juga sebagai mip6d merupakan perangkat lunak yang digunakan untuk membuat konfigurasi pada *Home Agent*, *Correspondent Node*, dan *Mobile Node* untuk menciptakan *environment Mobile* IPv6. Pengaturan *Route Optimization* dan *Bidirectional Tunneling* juga diatur disini.

3. RADVD (*Router Advertisment Daemon*)

   RADVD merupakan sebuah perangkat lunak yang digunakan sebagai pengirim pesan *Router Advertisement* yang diinstall di *Home Agent* dan *Foreign Agent* yang menjadi *Router*. Tujuannya adalah agar dapat memberikan IP pada *Mobile Node* yang meminta *router solicitation* pada *Home Agent* dan *Foreign Agent*.

4. Wing FTP *Server*

   Wing FTP *Server* adalah sebuah aplikasi berbasis web yang digunakan sebagai *server* FTP. Aplikasi ini mendukung sejumlah *File Transfer Protocol*, seperti FTP, HTTP, FTPS, HTTPS and SFTP dan memberikan fleksibilitas *end user* bagaimana terhubung ke *server*. Aplikasi berbasis web ini dapat diakses dari mana saja, memonitor performansi *server* dan *online session* dan menerima notifikasi *email* mengenai berbagai even yang terjadi di *server*. [6]

5. *Wireshark* 1.8.2

   Wireshark merupakan sebuah perangkat lunak yang digunakan untuk mengontrol *interface* jaringan agar *traffic* jaringan dapat terlihat. Aplikasi ini digunakan untuk membaca paket-paket yang dikirim pada *interface* jaringan. Parameter *transfer time*, *delay* dan *throughput* dapat dilihat di perangkat lunak ini dengan cara melakukan filter ke data yang ingin dilihat. Ringkasan paket-paket tersebut dapat dilihat di *Wireshark* Summary.

6. Xenotix Hash DOS Tester

Tools ini digunakan untuk melakukan sejumlah *flood* dengan cara mengatur berapa *thread* yang akan digunakan untuk melakukan penyerangan. Selain itu dapat juga mengirim sejumlah besar request dengan cara mengatur berapa besar request size dari paket yang akan digunakan untuk melakukan penyerangan. Ketika warna kuning muncuk di blok-blok *tools* ini, maka *tools* ini telah mengirim sejumlah paket yang telah diatur. Sementara warna merah yang muncul di blok-blok *tools*, *Node* yang dijadikan target akan mengkonsumsi penggunaan CPU sebesar 100%. *Tools* ini juga dapat melakukan infinite loop, sehingga serangan akan terjadi secara terus menerus. [7]

7. *Traffic* Generator

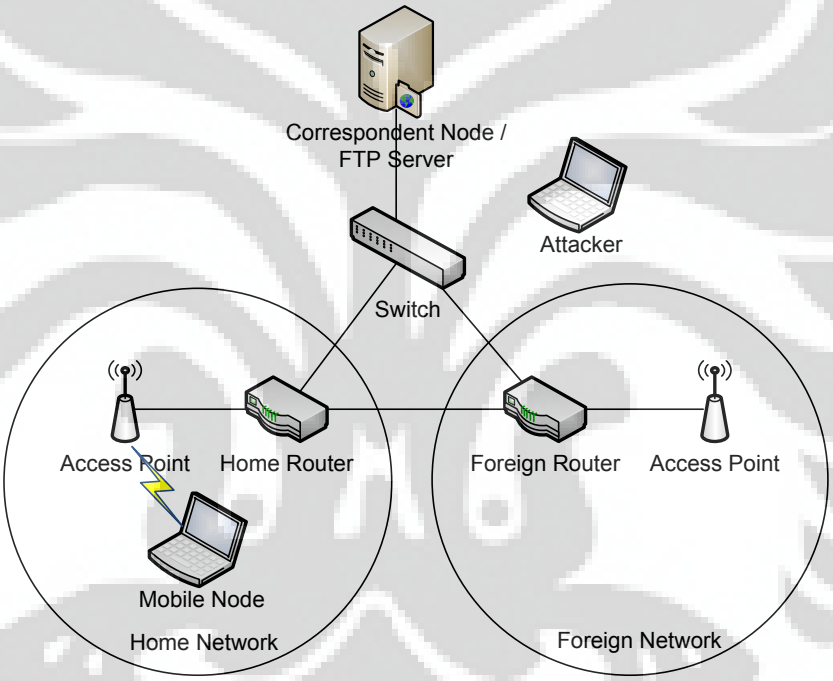
Traffic Generator adalah sebuah *tools* yang digunakan untuk membuat jaringan skala kecil dapat menyerupai jaringan sebenarnya yang terkoneksi dengan *traffic* yang padat. *Tools* ini dipasang pada *Node* dalam jaringan yang dibangun dan memberikan *traffic* pada jaringan. *Tools* yang digunakan adalah Multi-Generator (MGEN).

3.3 Skenario Penyerangan dengan *Threat*

Penyerangan jaringan *Mobile* IPv6 akan dilakukan dengan 4 macam skenario. Skenario pertama yaitu kondisi normal jaringan sebelum dilakukan penyerangan apapun ketika *Mobile Node* berada di *Home Network*. Skenario kedua yaitu kondisi normal jaringan sebelum dilakukan penyerangan apapun ketika *Mobile Node* berada di *Foreign Network*. Skenario ketiga yaitu dengan penyerangan menggunakan metode *Denial of Service* ketika *Mobile Node* berada di *Home Network*. Skenario keempat yaitu dengan penyerangan menggunakan metode *Denial of Service* ketika *Mobile Node* berada di *Foreign Network*.

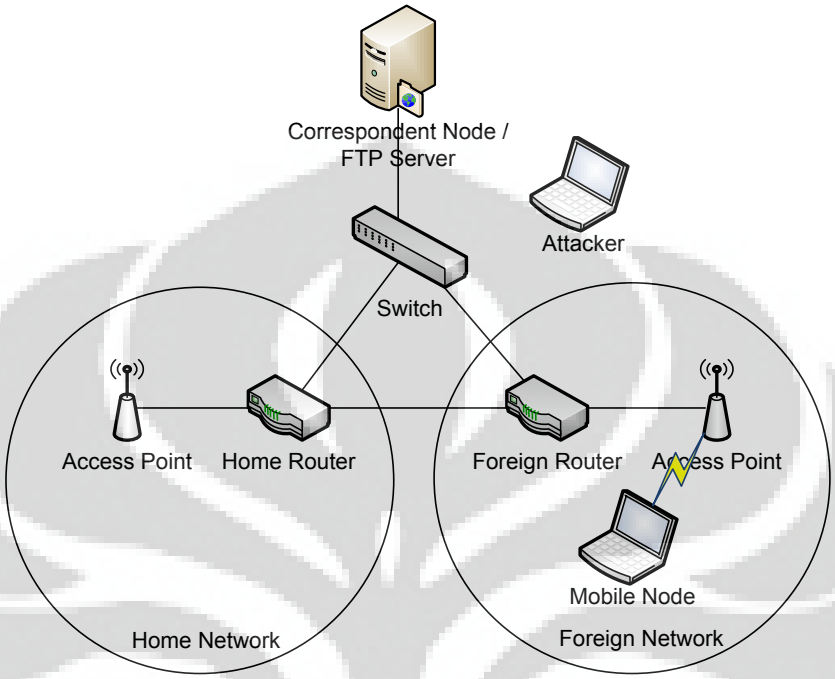
Alasan penggunaan *Route Optimization* sebagai metode *routing* yang digunakan untuk penelitian pada topologi ini adalah karena metode *routing* ini merupakan metode yang digunakan pada *Mobile* IPv6 sekarang ini. *Bidirectional Tunneling* merupakan teknologi yang dulu digunakan untuk *Mobile* IPv6 pada tahap awal pengembangan. Alasan lain penggunaan *Route Optimization* adalah karena metode ini lebih baik daripada *Bidirectional Tunneling* yang mengharuskan paket data melewati *Home Agent* terlebih dahulu untuk dapat diteruskan ke *Mobile Node* dari *Correspondent Node*.

Skenario pertama akan dilakukan pada jaringan seperti pada Gambar 3.2.



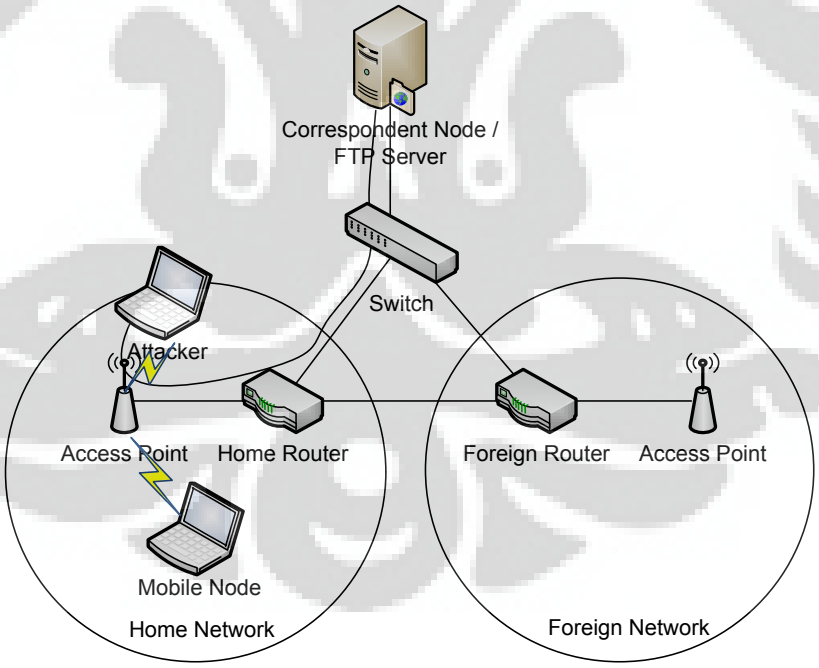
Gambar 3.2 *Mobile Node* di *Home Network* sebelum dilakukan penyerangan

Skenario kedua akan dilakukan pada jaringan seperti pada Gambar 3.3.



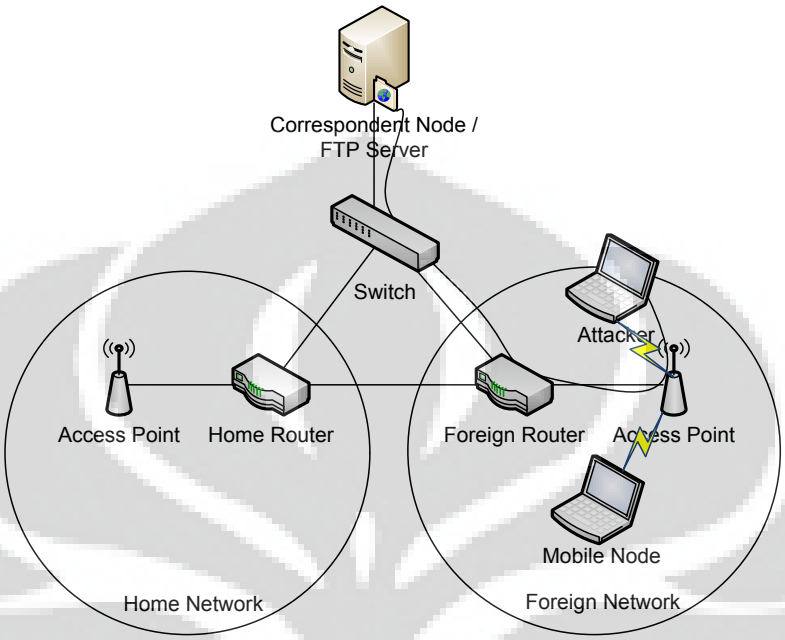
Gambar 3.3 *Mobile Node* di *Foreign Network* sebelum dilakukan penyerangan

Skenario ketiga akan dilakukan pada jaringan seperti pada Gambar 3.4.



Gambar 3.4 Serangan dengan *Denial of Service* ketika *Mobile Node* berada di *Home Network*

Skenario keempat akan dilakukan pada jaringan seperti pada Gambar 3.5.



Gambar 3.5 Serangan dengan *Denial of Service* ketika *Mobile Node* berada di *Foreign Network*

Penyerangan dengan metode ini akan melumpuhkan sistem dengan objek yang diserang adalah FTP *Server*. Penyerang mengirimkan *flood* paket ICMP dengan ukuran lebih dari 65 KB ke FTP *server*. Hasilnya adalah akan terjadi *crash* pada *operating system*. Contohnya seperti pada OS Linux versi di bawah 2.0.23 hasilnya adalah kernel menjadi panik. Pada saat yang bersamaan *Mobile Node* mencoba mengunduh sebuah data dari FTP *server*. *Transfer time*, *delay* dan *throughput* akan diamati ketika penyerangan ini terjadi. Pengambilan data nantinya akan dilakukan 10 kali pada saat *Mobile Node* berada di *Home Network* dan 10 kali pada saat *Mobile Node* berada di *Foreign Network*. Hasil pada saat berada di *Home Network* dan *Foreign Network* selanjutnya akan dibandingkan dan dianalisis. Serangan akan menggunakan Xenotix Hash DOS Tester.

Threat yang akan digunakan untuk menyerang berbeda - beda. Pertama akan digunakan 12 *thread*. Yang kedua akan digunakan 18 *thread* dan yang ketiga akan digunakan 24 *thread*. Hal ini dilakukan untuk melihat sejauh mana perbedaan

performansi jaringan yang akan didapat dengan memvariasikan jumlah *threat* yang digunakan untuk menyerang jaringan tersebut. Sementara ukuran paket yang digunakan untuk melakukan serangan adalah tetap, yaitu sebesar 600 KB.

3.4 Penyusunan dan Instalasi Sistem

Setelah topologi jaringan selesai dibuat, selanjutnya sistem yang digunakan harus disesuaikan dengan perangkat keras yang telah disiapkan agar dapat menciptakan suatu lingkungan jaringan yang mendukung *Mobile* IPv6.

3.4.1. Instalasi Kernel

Sistem operasi Linux Ubuntu 13.04 LTS (*Raring Ringtail*) membutuhkan instalasi tambahan kernel agar dapat mendukung fitur *Mobile* IPv6. Kernel yang digunakan adalah kernel Linux-3.8.2 yang mengaktifkan fitur *Mobile* IPv6 tersebut. Kernel Linux-3.8.2 dapat diunduh di:

http://kambing.ui.ac.id/linux/v3.x/linux-3.8.2.tar.bz2

File linux-3.8.2.tar.bz2 disimpan di dalam direktori /usr/src. Untuk mengunduh *file* tersebut ke dalam direktori /usr/src, harus menggunakan terminal.

3.4.2. Instalasi UMIP dan RADVD

Environment Mobile IPv6 dapat diciptakan dengan cara melakukan penyusunan terhadap paket-paket yang terdapat pada UMIP. Agar UMIP dapat berjalan dan digunakan, terdapat paket-paket perangkat lunak yang harus diinstalasi terlebih dahulu seperti autoconf, automake, bison, flex, libssl-dev, indent, ipsec-*tools*, dan radvd.

3.4.3. Instalasi Wing FTP *Server*

Di dalam sistem FTP dibutuhkan sebuah FTP *server* dan FTP klien untuk dapat melakukan *transfer file*. Dalam pembuatan sistem FTP ini, digunakan Wing FTP *server* yang dapat berfungsi sebagai *server* sekaligus klien. Wing FTP *Server* ini berbasis web sehingga user yang ingin melakukan *transfer file* harus membuat akun dan login terlebih dahulu sebelum dapat melakukan *transfer file*. Wing FTP *Server* ini dapat diunduh di:

www.wftp*server*.com/download.htm

Di halaman web Wing FTP *Server* tersebut pilih Wing FTP *Server* v4.2.0 for Linux 32-bit with kernel 2.6.x. Lalu unduh *file* tersebut ke PC. *File* yang telah diunduh masih berada dalam format gzipped tar. Ekstrak *file* tersebut dengan cara melakukan perintah sebagai berikut di dalam terminal:

```
#tar xzvf wftpserver 2.6.x.tar.gz
```

Perintah tersebut akan mengesktrak *file* dari package ke dalam direktori (direktori instalasi) bernama "wftp*server*". Selanjutnya masuk ke dalam direktori wftp*server* tersebut dengan melakukan perintah:

```
#cd wftpserver
```

Lalu lakukan *setup* Wing FTP *Server* dengan perintah:

```
#sudo ./setup.sh
```

Ketika instalasi telah selesai dilakukan, maka Wing FTP *Server* dapat dibuka dari web browser dimanapun dengan mengetik [2001:db8:ffff:100b::13]:5466 (IP tersebut merupakan IP yang diberikan kepada *server* FTP). 5466 adalah *default port* yang diberikan untuk *server*. *Port* tersebut dapat diganti ketika melakukan instalasi Wing FTP *Server*.

3.4.4. Rekayasa Trafik

Perangkat lunak yang digunakan untuk melakukan rekayasa trafik ini adalah MGEN atau dikenal juga sebagai *Multi Generator*. Perangkat lunak ini dapat membuat pola trafik pada jaringan sehingga membuat jaringan memiliki beban dengan cara yang bervariasi.

MGEN memerlukan *server* agar dapat mengirimkan paket-paket yang diminta dan client yang berperan mendengar permintaan koneksi, merespon dan menerima paket yang dikirim.

MGEN dapat dijalankan dengan cara mengetikkan perintah berikut di terminal:

```
#mgen [ipv6] [input <scriptfile>] [output <logfile>]
```

`[ipv6]` merupakan protokol yang digunakan adalah IPv6, `[input <scriptfile>]` merupakan *script file* yang berisi perintah untuk membuat trafik MGEN, `[output <logfile>]` merupakan perintah ke MGEN untuk mencatat perintah yang dijalankan dalam sebuah log *file*. Pada `<scriptfile>` diganti dengan direktori dimana script *file* tersebut berada dan `<logfile>` diganti dengan direktori dimana log *file* tersebut ingin disimpan.

MGEN digunakan pada *Home Agent, Foreign Agent,* dan *Correspondent Node*. Isi script *file* untuk masing-masing *Node* berbeda - beda. Pada *Home Agent* berisi script untuk mengirim pesan UDP sebanyak 32 pesan per detik dengan ukuran 8192 bytes dari *port* 5001 ke *port* 5001 *Foreign Agent*. *Foreign Agent* juga mengirim pesan sebanyak 32 pesan per detik dengan ukuran 8192 bytes dari *port* 5000 ke *port* 5000 *Home Agent*. Sementara itu *Correspondent Node* mendengarkan pesan UDP dari *Foreign Agent* di *port* 5002, mendengarkan pesan TCP dari *Home Agent* di *port* 8000.

Home Agent mengirim pesan TCP secara periodik sebanyak 1 pesan per detik dengan ukuran sebesar 5242880 bytes. *Foreign Agent* mengirim pesan UDP secara periodik sebanyak 32 pesan per detik dengan ukuran sebesar 8192.

3.4.5. Konfigurasi *Node*

1. *Home Agent*

Home Agent memiliki 2 *interface* yang digunakan untuk jaringan yang dibuat, yaitu *interface* eth0 dan *interface* eth1. *Interface* eth0 digunakan untuk menghubungkan *Home Agent* dengan *Switch* dengan alamat IP 2001:db8:ffff:100b::11/64. *Interface* eth1 digunakan untuk menghubungkan *Home Agent* dengan *Access Point* di *Home Network* dengan alamat IP 2001:db8:ffff:100a::1/64.

Kedua *interface* tersebut dikonfigurasi terlebih dahulu agar *interface* tersebut dapat digunakan di *Home Agent*. *Home Agent* yang berfungsi sebagai *router* juga perlu diaktifkan terlebih dahulu fiturnya. Static routing ditambahkan agar *Home Agent* dapat berkomunikasi dengan *Foreign Agent*.

Pada *Home Agent* ini juga dibutuhkan dukungan konfigurasi *Node* berupa UMIP Linux *Mobile* IPv6 untuk mengaktifkan fitur *Mobile* IPv6. Konfigurasi ini disimpan di dalam direktori /usr/local/etc dengan nama *file* mip6d.conf. Terakhir, *Home Agent* membutuhkan RADVD untuk mengaktifkan *router Advertisement*.Konfigurasi ini disimpan di dalam direktori /etc dengan nama *file* radvd.conf.

2. *Foreign Agent*

Foreign Agent memiliki 2 *interface* yang digunakan untuk jaringan yang dibuat, yaitu *interface* eth0 dan eth1. *Interface* eth0 digunakan untuk menghubungkan *Foreign Agent* dengan *Switch* dengan alamat 2001:db8:ffff:100b::12/64. *Interface* eth1 digunakan untuk menghubungkan *Foreign Agent* dengan *Access Point* di *Foreign Network* dengan alamat 2001:db8:ffff:100c::1/64.

Kedua *interface* tersebut dikonfigurasi terlebih dahulu agar *interface* tersebut dapat digunakan di *Foreign Agent*. *Foreign Agent* yang berfungsi

sebagai *router* juga perlu diaktifkan terlebih dahulu fiturnya. Static routing ditambahkan agar *Foreign Agent* dapat berkomunikasi dengan *Home Agent*.

Pada *Foreign Agent* ini tidak dibutuhkan dukungan konfigurasi *Node* UMIP Linux *Mobile* IPv6. Hal ini dikarenakan *Foreign Agent* tidak perlu menjaga koneksi dengan *Mobile Node*. Terakhir, *Foreign Agent* membutuhkan RADVD untuk mengaktifkan *router Advertisement*. Konfigurasi ini disimpan di dalam direktori /etc dengan nama *file* radvd.conf.

3. Corresspondent *Node*

Correspondent Node hanya memiliki 1 *interface*, yaitu *interface* eth0. *Interface* eth0 ini digunakan untuk menghubungkan ke *Switch* yang juga terhubung dengan *Home Agent* dan *Foreign Agent*. Alamat IP yang digunakan pada *interface* eth0 ini adalah 2001:db8:ffff:100b::13/64.

Pada *Correspondent Node* ini juga diperlukan konfigurasi static routing agar dapat terhubung dengan *Home Agent* dan *Foreign Agent*. *Correspondent Node* menggunakan UMIP Linux *Mobile* IPv6 juga seperti pada *Home Agent*.

4. *Mobile Node*

Mobile Node menggunakan konfigurasi mip6d.conf untuk mengaktifkan fitur *Mobile* IPv6. Konfigurasi UMIP Linux *Mobile* IPv6 juga digunakan bersamaan dengan konfigurasi mip6d.conf. Konfigurasi pada *Mobile Node* tidak diatur secara manual, tetapi ada beberapa fitur yang juga perlu diatur.

BAB 4

PENGUJIAN DAN ANALISA PARAMETER PERFORMANSI JARINGAN *MOBILE* IPV6 DENGAN APLIKASI FTP

4.1 Pengujian Jaringan *Mobile* IPv6 Pada Aplikasi FTP

Jaringan *Mobile* IPv6 yang dibuat terdiri dari *Mobile Node, Correspondent Node* dan *Router*. Jaringan ini membutuhkan trafik seperti aslinya sehingga digunakan perangkat lunak Trafik Generator yang berfungsi sebagai pemberi trafik yang menyerupai jaringan asli.

Pengujian jaringan tersebut menggunakan *Wireshark* yang dapat menampilkan paket data yang lewat pada jaringan dan dapat diamati statistiknya. Data yang diamati adalah paket-paket FTP yang lewat di jaringan. Pengujian dilakukan dengan cara melakukan pengunduhan (download) *file* dari FTP *Server* ke *Mobile Node* yang selanjutnya akan dilakukan penyerangan kepada *Correspondent Node* jaringan tersebut.

4.2 Pengukuran Parameter Performansi

Pada bab sebelumnya telah dijelaskan bahwa parameter pengukuran jaringan *Mobile* IPv6 yang akan digunakan adalah *transfer time*, *delay* dan *throughput*. Metode yang digunakan yaitu *Route Optimization* dengan serangan yang akan digunakan adalah *Denial of Service* yang selanjutnya akan dilakukan 4 (empat) skenario pengukuran.

Sesuai dengan topologi jaringan yang telah dibuat, performansi yang diukur adalah *Mobile Node* yang melakukan perpindahan, yaitu ketika berada di *Home Network* dan *Foreign Network*. Dari 2 macam *network* tersebut selanjutnya dibagi lagi menjadi 2 macam metode, yaitu pengukuran sebelum dilakukan penyerangan, dan pengukuran dengan serangan *Denial of Service*.

Dari kedua macam metode tersebut akan didapatkan 4 macam skenario pengukuran, yaitu pengukuran sebelum penyerangan di *Home Network*, pengukuran

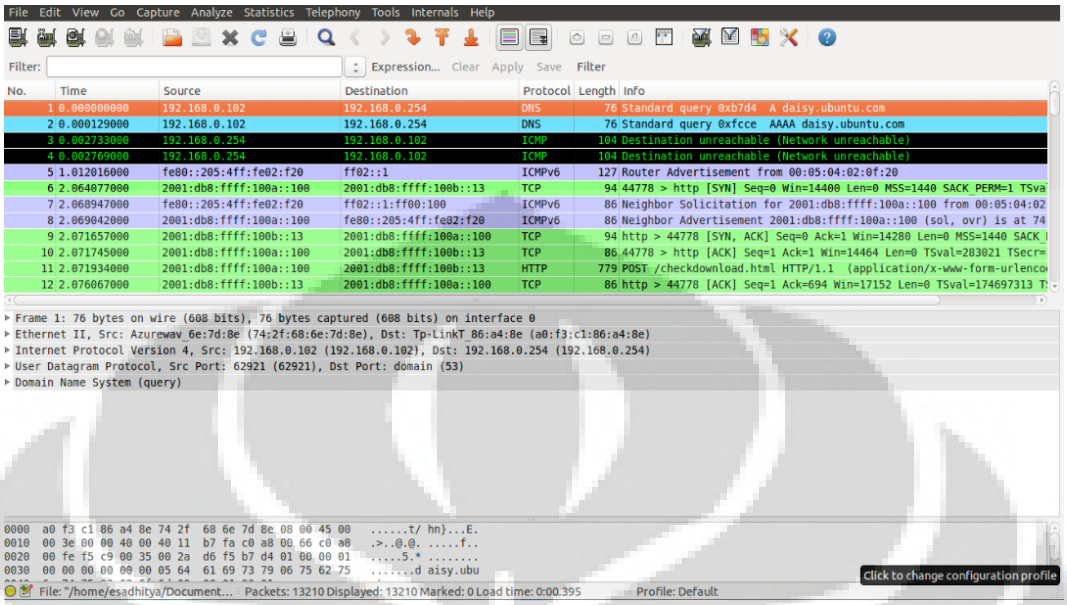
sebelum penyerangan di *Foreign Network*, pengukuran dengan penyerangan *Denial of Service* di *Home Network*, dan pengukuran dengan penyerangan *Denial of Service* di *Foreign Network*.

Dari empat macam skenario tersebut, *Mobile Node* akan melakukan pengunduhan *file* yang berada di *Correspondent Node* / FTP *Server* dengan aplikasi FTP dan performansi jaringan akan diukur saat dilakukan pengiriman dari *Correspondent Node* ke *Mobile Node*. Pengukuran akan dilakukan sebanyak 10 kali.

Dalam melakukan pengunduhan, akan digunakan sebuah *file*, yaitu *file* .tar.bz yang berukuran 11.75 MB. Pengukuran akan dilakukan sebanyak 10 kali untuk masing-masing skenario yang ada.

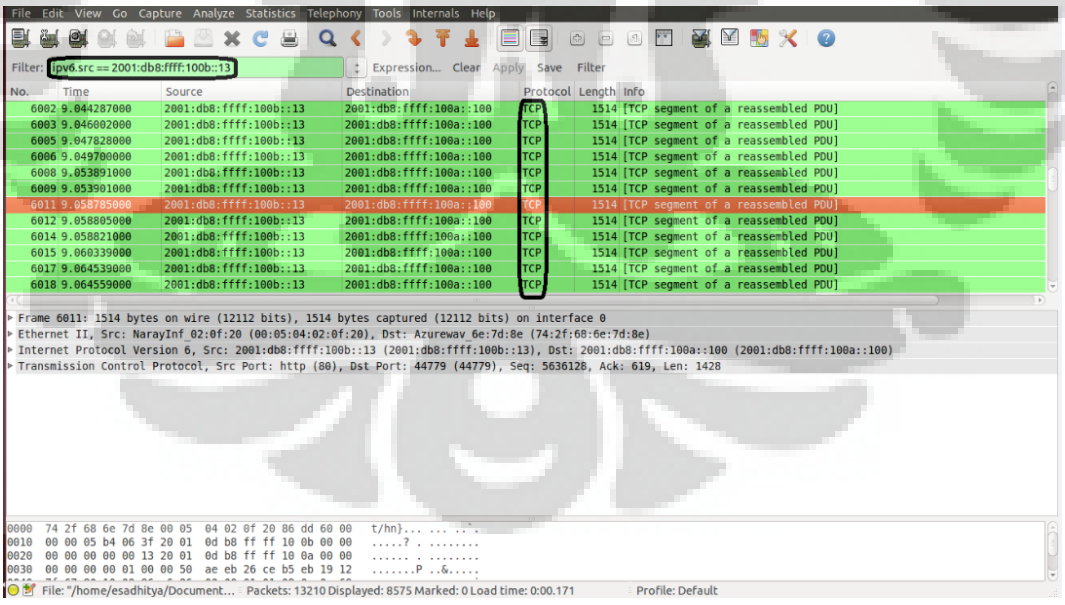
Dari serangan *Denial of Service*, penyerangan akan mengkombinasikan *thread* yang digunakan, sementara ukuran paket tetap. Pertama yaitu 12 *thread* dengan ukuran paket sebesar 600 KB. Kedua yaitu 18 *thread* dengan ukuran paket 600 KB. Ketiga yaitu 24 *thread* dengan ukuran paket 600 KB.

Selanjutnya dengan menggunakan *Wireshark* dapat dilakukan *capture* terhadap paket-paket data yang dikirim dari *Correspondent Node* ke *Mobile Node* dalam setiap pengunduhan *file* yang dilakukan. Dari capture tersebut akan didapatkan beberapa data parameter seperti *transfer time*, *delay* dan *throughput* yang selanjutnya dapat dianalisa. Gambar 4.1 adalah tampilan *Wireshark* pada saat menangkap paket-paket yang diterima.



Gambar 4.1 Tampilan *Wireshark* Sebelum dilakukan Filter Data

Untuk menampilan paket-paket data yang berasal dari *Correspondent Node*, dilakukan filter dengan cara memasukkan "ipv6.src == 2001:db8:ffff:100b::13" ke dalam menu Filter. Gambar 4.2 mengilustrasikan *Wireshark* setelah dilakukan filterisasi data.



Gambar 4.2 Tampilan *Wireshark* Setelah Filter Data TCP di *Correspondent Node*

Setelah dilakukan filter data, nilai-nilai yang dijadikan parameter yang akan dianalisa akan didapatkan dengan cara menampilkan *Summary* dari proses pengiriman data. Gambar 4.3 merupakan tampilan Summary dari *Wireshark*.

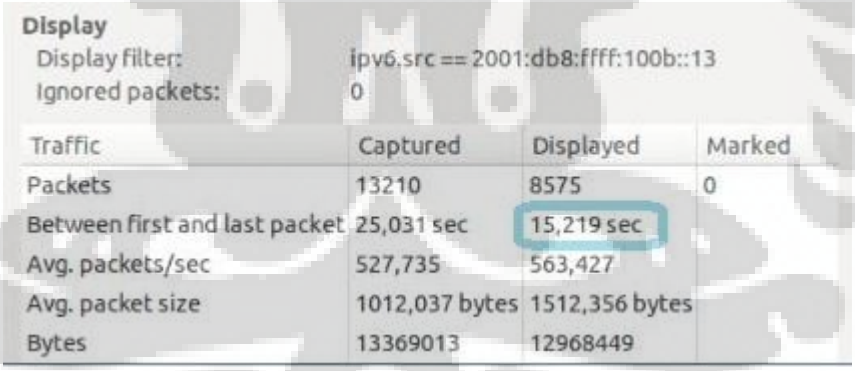
Gambar 4.3 Tampilan Summary pada *Wireshark*

Dalam Summary *Wireshark* terdapat nilai-nilai dari paket-paket tersebut. Kolom "*Displayed*" dan baris "*Between first and last packet*" merupakan *transfer time*, kolom "*Displayed*" dan baris "*Avg. packet size*" merupakan *throughput*, kemudian nilai *transfer time* dibagi dengan nilai pada kolom "*Displayed*" dan baris "*Packets*" merupakan *delay*.

4.3 Analisa Parameter Performansi

4.3.1. Analisa *Transfer time*

Transfer time adalah sebuah parameter yang digunakan untuk menunjukkan berapa lama waktu yang dibutuhkan dari paket pertama dikirimkan hingga paket yang terakhir dikirimkan. Cara menghitung *transfer time* adalah dengan melihat awal waktu pertama kali paket dikirimkan hingga akhir waktu paket terakhir dikirimkan. Analisa *transfer time* dilakukan berdasarkan posisi *Mobile Node* ketika berada di *Home Network* dan *Foreign Network*, baik pada saat sebelum dilakukan penyerangan dan pada saat dilakukan penyerangan *Denial of Service*. Pada Summary *Wireshark*, *transfer time* dapat dilihat pada bagian yang diberi kotak biru pada Gambar 4.4.



Display

Display filter: ipv6.src == 2001:db8:ffff:100b::13
Ignored packets: 0

Traffic	Captured	Displayed	Marked
Packets	13210	8575	0
Between first and last packet	25,031 sec	15,219 sec	
Avg. packets/sec	527,735	563,427	
Avg. packet size	1012,037 bytes	1512,356 bytes	
Bytes	13369013	12968449	

Gambar 4.4 *Transfer time* pada Summary *Wireshark*

a. *Transfer time* Sebelum *Network* Diserang

Pertama kali data yang diambil pada analisa *transfer time* ini adalah pada saat *Mobile Node* berada di *Home Network* dan *Foreign Network*. *Mobile Node* mengunduh *file* sebesar 11.75 MB sebanyak 10 kali di masing-masing *network* tersebut. Dari hasil pengunduhan tersebut didapatkan data rata-rata *transfer time* seperti pada Tabel 4.1

Tabel 4.1 Rata-rata *transfer time* sebelum diserang

Pengukuran ke -	*Transfer time* (s)	
	Home Network	*Foreign Network*
1	15.173	15.219
2	15.143	15.202
3	15.18	15.221
4	15.859	15.214
5	15.143	15.32
6	15.202	15.251
7	15.069	15.27
8	15.15	15.25
9	15.115	15.441
10	15.152	15.236
Rata-rata	15.219	15.262

Berdasarkan Tabel 4.1 dapat dilihat perbedaan *transfer time* pada saat *Mobile Node* berada di *Home Network* dan *Foreign Network*. Besar data yang diunduh sama, perbedaan *transfer time*nya tidak berbeda jauh walaupun *Mobile Node* sudah berpindah *network*.

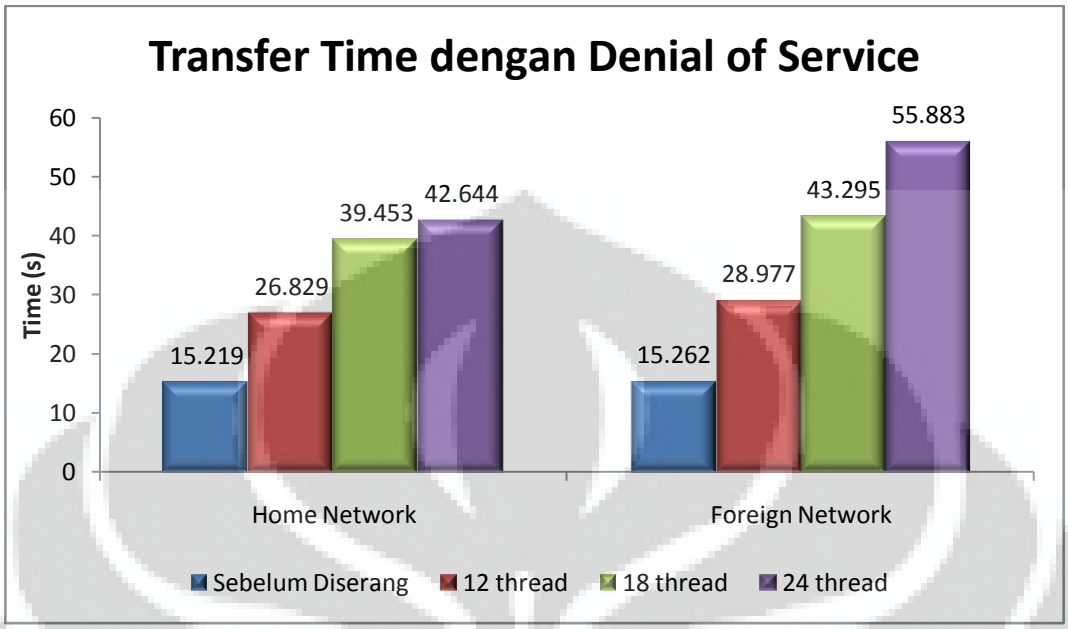
b. *Transfer time* dengan serangan *Denial of Service*

Pengambilan data yang kedua yaitu ketika *Mobile Node* melakukan pengunduhan di *Home Network* dan *Foreign Network*. *Mobile Node* mengunduh *file* sebesar 11.75 MB sebanyak 10 kali. Pada saat yang sama dilakukan penyerangan ke FTP *Server* dengan metode *Denial of Service*. Selanjutnya didapatkan rata-rata *transfer time* seperti pada Tabel 4.2

Tabel 4.2 Rata-rata *transfer time* dengan Serangan *Denial of Service*

Pengukuran ke-	*Transfer time* (s)					
	Home Network			*Foreign Network*		
	12	18	24	12	18	24
	(*thread*)					
1	26.015	43.741	43.919	28.294	36.705	42.627
2	25.437	34.141	43.188	30.221	37.426	56.718
3	24.855	36.322	41.682	27.881	36.705	57.323
4	27.658	48.244	40.489	26.971	48.105	57.932
5	27.542	38.441	45.706	29.825	57.073	39.438
6	28.142	33.252	49.125	27.182	48.18	65.633
7	27.268	36.722	44.103	28.81	39.968	80.458
8	28.133	49.401	38.478	29.928	58.165	75.65
9	26.33	37.752	38.703	27.791	35.738	43.518
10	26.907	36.513	41.043	32.865	34.883	39.532
Rata-rata	26.829	39.453	42.644	28.977	43.295	55.883

Dari Tabel 4.2 dapat dilihat bahwa nilai *transfer time* semakin meningkat dan berbanding lurus dengan banyaknya *thread* yang digunakan untuk menyerang. Gambar 4.5 menampilkan grafik perbandingan rata-rata *transfer time Mobile Node* saat mengunduh *file* di *Home Network* dan *Foreign Network* dengan serangan *Denial of Service*.



Gambar 4.5 *Transfer time* dengan *Denial of Service*

Jumlah *thread* yang berbeda berpengaruh secara signifikan karena semakin banyak *thread* yang diberikan, maka *transfer time* juga akan semakin bertambah sesuai dengan banyaknya *thread* yang diberikan kepada *Correspondent Node*. Peningkatan *transfer time* dapat dihitung dengan cara:

$$\frac{transfer\ \ time\ \ dengan\ \ thread - transfer\ \ time\ \ sebelum\ \ diserang}{transfer\ \ time\ \ sebelum\ \ diserang} \text{ x } 100\ \%$$

Di *Home Network, Denial of Service* sebanyak 12 *thread* mengakibatkan *transfer time* meningkat sebesar 76.28% bila dibandingkan dengan sebelum dilakukan penyerangan. *Denial of Service* sebanyak 18 *thread* mengakibatkan *transfer time* meningkat sebesar 159.23% bila dibandingkan dengan sebelum dilakukan penyerangan. *Denial of Service* sebanyak 24 *thread* mengakibatkan *transfer time* meningkat sebesar 180.2% bila dibandingkan dengan sebelum dilakukan penyerangan.

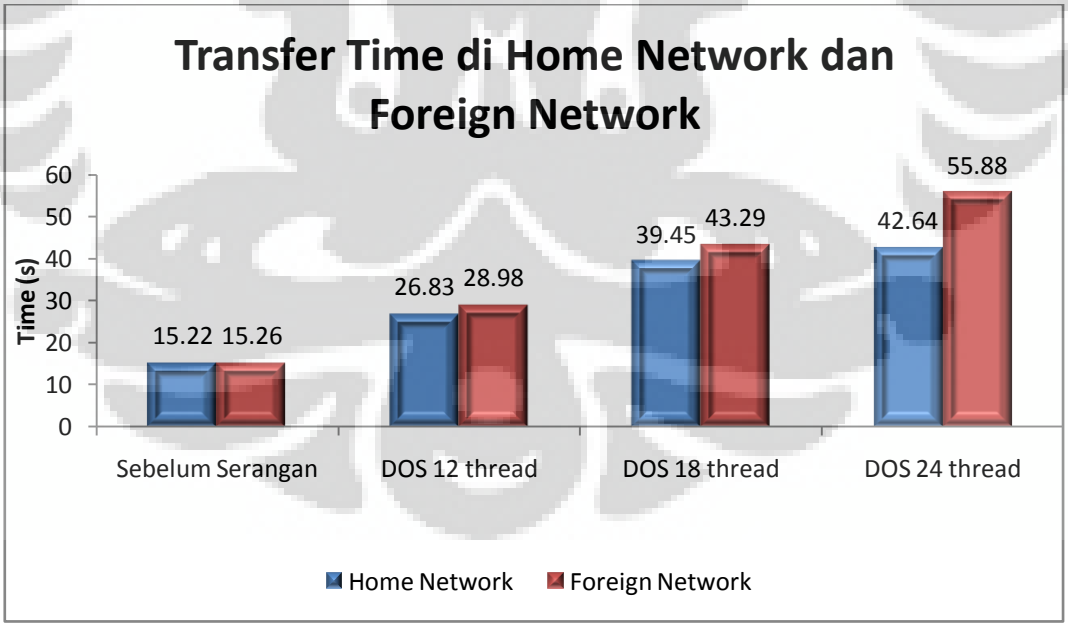
Sementara itu pada *Foreign Network, Denial of Service* sebanyak 12 *thread* mengakibatkan *transfer time* meningkat sebesar 89.86% bila dibandingkan dengan

sebelum dilakukan penyerangan. *Denial of Service* sebanyak 18 *thread* mengakibatkan *transfer time* meningkat sebesar 183.67% bila dibandingkan dengan sebelum dilakukan penyerangan. *Denial of Service* sebanyak 24 *thread* mengakibatkan *transfer time* meningkat sebesar 266.16% bila dibandingkan dengan sebelum dilakukan penyerangan.

c. Perbandingan *Transfer Time* antara *Home Network* dan *Foreign Network*

Analisa perbandingan *transfer time* di *Home Network* dan *Foreign Network* dibuat dengan menggunakan rata-rata nilai *transfer time* di *Home Network* dan *Foreign Network*. Berdasarkan rata-rata data yang didapat, nilai *transfer time* cenderung lebih kecil ketika berada di *Home Network*.

Transfer time terkecil didapat pada saat skenario tanpa menggunakan serangan di *Home Network* dan *transfer time* terbesar didapat pada saat skenario dengan menggunakan serangan di *Foreign Network*. Gambar 4.6 memperlihatkan perbandingan grafik rata-rata *transfer time* untuk masing-masing skenario di *Home Network* dan *Foreign Network*.



Gambar 4.6 *Transfer time* di *Home Network* dan *Foreign Network*

Dari grafik *transfer time* tersebut, terdapat 2 kondisi yang dapat dibandingkan, yaitu ketika berada di *Home Network* dan *Foreign Network*. Perbandingan kecepatan *transfer time* antara *Home Network* dan *Foreign Network* ini dapat dihitung dengan cara:

$$\frac{trans\,fer\ time\ \ Foreign\ \ Network\ -tranfer\ \ time\ \ Home\ \ Network}{tranfer\ \ time\ \ Home\ \ Network} \text{ x } 100\%$$

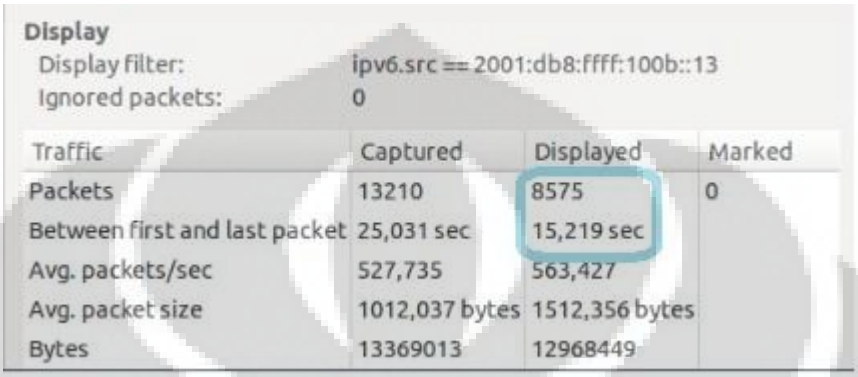
Pertama yaitu *transfer time* di *Home Network* lebih cepat daripada di *Foreign Network* sebesar 0.28%. Kedua yaitu *transfer time* di *Home Network* lebih cepat daripada di *Foreign Network* sebesar 8.01% untuk serangan dengan 12 *thread* pada *Denial of Service*. Ketiga yaitu *transfer time* di *Home Network* lebih cepat daripada di *Foreign Network* sebesar 9.73% untuk serangan dengan 18 *thread* pada *Denial of Service.* Keempat yaitu *transfer time* di *Home Network* lebih cepat daripada di *Foreign Network* sebesar 31.05% untuk serangan dengan 24 *thread* pada *Denial of Service.*

Hal ini dapat terjadi karena *Mobile Node* menggunakan metode *Route Optimization* yang menggunakan ipv6 *tunnel*, yang menghubungkan *Mobile Node* langsung ke Correspondont *Node* sehingga data tidak harus melewati *Home Agent* terlebih dahulu. Dengan *tunnel* yang disediakan *Route Optimization* ini, *Mobile Node* seolah-olah tetap berada di *Home Network* walaupun sudah berpindah jaringan ke *Foreign Network.*

4.3.2. Analisa *Delay*

Delay adalah sebuah parameter yang dapat digunakan untuk menunjukkan waktu yang dibutuhkan bagi sebuah paket dari saat dimulai pengiriman hingga sampai ke tujuan. Cara menghitung *delay* adalah dengan cara melakukan pembagian dari waktu total pengiriman dengan jumlah paket pada sebuah *file* tersebut. Analisa pada *delay* dilakukan berdasarkan posisi *Mobile Node* ketika berada di *Home Network* dan *Foreign Network,* baik pada saat sebelum dilakukan penyerangan dan

pada saat dilakukan penyerangan dengan *Denial of Service*. Pada Summary *Wireshark, delay* dapat dilihat pada bagian yang diberi kotak biru pada Gambar 4.7.



Display

Display filter: ipv6.src == 2001:db8:ffff:100b::13

Ignored packets: 0

Traffic	Captured	Displayed	Marked
Packets	13210	8575	0
Between first and last packet	25,031 sec	15,219 sec	
Avg. packets/sec	527,735	563,427	
Avg. packet size	1012,037 bytes	1512,356 bytes	
Bytes	13369013	12968449	

Gambar 4.7 *Delay* pada Summary *Wireshark*

Besarnya nilai *delay* didapat dengan cara membagi nilai antara kolom "Displayed dan baris "Between first and last packet" dengan kolom "Displayed" dan baris "Packets".

a. *Delay* pada *Network* Sebelum Diserang

Pertama kali data yang diambil pada analisa *delay* ini adalah pada saat *Mobile Node* berada di *Home Network* dan *Foreign Network*. *Mobile Node* mengunduh *file* sebesar 11.75 MB sebanyak 10 kali di masing-masing *network* tersebut. Dari hasil pengunduhan tersebut didapatkan data rata-rata *delay* seperti pada Tabel 4.3

Tabel 4.3 Rata-rata *delay* pada *Network* sebelum diserang

Pengukuran ke -	*Delay* (ms)	
	Home Network	*Foreign Network*
1	1.759	1.774
2	1.754	1.779
3	1.761	1.794
4	1.876	1.811
5	1.755	1.805
6	1.819	1.806
7	1.751	1.796
8	1.756	1.778
9	1.755	1.798
10	1.764	1.804
Rata-rata	1.775	1.795

Berdasarkan Tabel 4.3 dapat dilihat perbedaan *delay* pada saat *Mobile Node* berada di *Home Network* dan *Foreign Network*. Besar data yang diunduh sama, perbedaan *delay* tidak berbeda jauh walaupun *Mobile Node* sudah berpindah *network*.

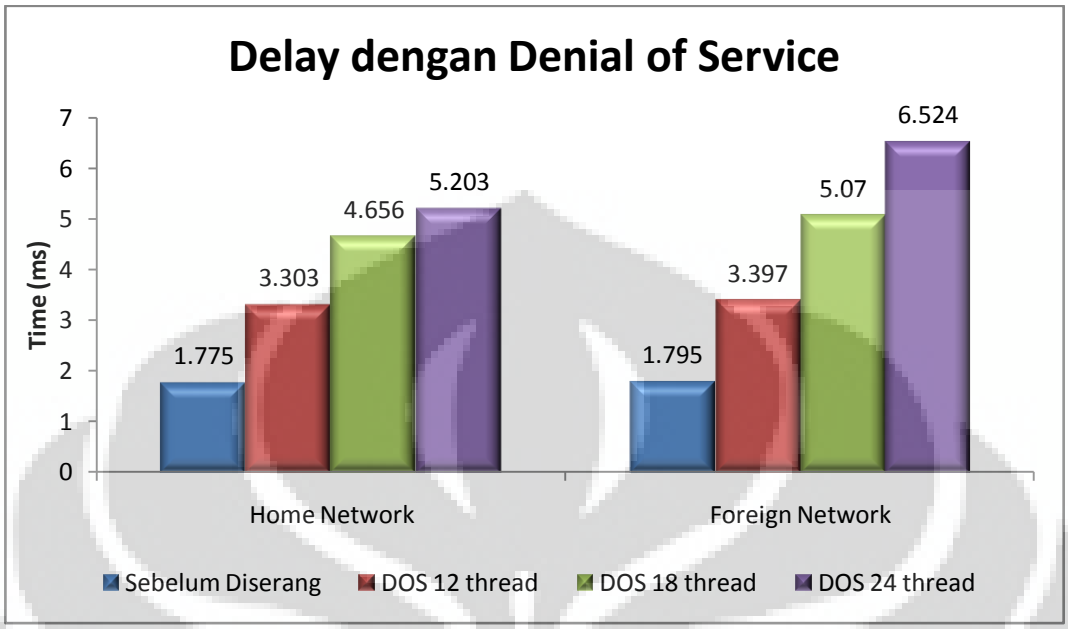
b. *Delay* dengan serangan *Denial of Service*

Pengambilan data yang kedua yaitu ketika *Mobile Node* melakukan pengunduhan di *Home Network* dan *Foreign Network*. *Mobile Node* mengunduh *file* sebesar 11.75 MB sebanyak 10 kali. Pada saat yang sama dilakukan penyerangan ke FTP *Server* dengan metode *Denial of Service*. Selanjutnya didapatkan rata-rata *delay* seperti pada Tabel 4.4

Tabel 4.4 Rata-rata *delay* pada *Home Network* dengan Serangan *Denial of Service*

Pengukuran ke-	*Delay* (ms)					
	Home Network			*Foreign Network*		
	12	18	24	12	18	24
	(*thread*)					
1	3.586	5.07	5.414	3.277	4.475	5.086
2	2.955	3.997	5.918	3.549	4.355	6.566
3	2.923	4.875	5.213	3.242	4.475	6.877
4	3.519	5.644	4.705	3.139	5.581	6.708
5	3.21	4.467	5.69	3.496	6.596	4.552
6	3.737	3.849	5.746	3.174	5.602	7.716
7	3.187	4.303	5.174	3.356	4.635	9.315
8	3.413	5.715	4.46	3.499	6.729	8.768
9	3.136	4.41	4.574	3.269	4.144	5.082
10	3.368	4.231	5.138	3.969	4.111	4.572
Rata-rata	3.303	4.656	5.203	3.397	5.070	6.524

Dari Tabel 4.4 dapat dilihat bahwa besar *delay* untuk tiap *thread* yang berbeda berpengaruh secara signifikan karena semakin banyak *thread* yang diberikan, maka *delay* juga akan semakin bertambah sesuai dengan banyaknya *thread* yang diberikan kepada *Correspondent Node*. Pengaruh serangan *Denial of Service* terhadap *delay* adalah bahwa terjadi peningkatan *delay* dibandingkan *Correspondent Node* tidak diserang. Gambar 4.8 menampilkan grafik perbandingan rata-rata *delay Mobile Node* saat mengunduh *file* di *Home Network* dan *Foreign Network* dengan serangan *Denial of Service*.



Gambar 4.8 *Delay* dengan *Denial of Service*

Jumlah *thread* yang berbeda berpengaruh secara signifikan karena semakin banyak *thread* yang diberikan, maka *delay* juga akan semakin bertambah sesuai dengan banyaknya *thread* yang diberikan kepada *Correspondent Node*. Peningkatan *delay* dapat dihitung dengan cara:

$$\frac{delay\ \ dengan\ \ thread - delay\ \ sebelum\ \ diserang}{delay\ \ sebelum\ \ diserang} \text{ x } 100\ \%$$

Di *Home Network*, *Denial of Service* sebanyak 12 *thread* mengakibatkan *delay* meningkat sebesar 86.08% bila dibandingkan dengan sebelum dilakukan penyerangan. Sebanyak 18 *thread* mengakibatkan *transfer time* meningkat sebesar 162.3% bila dibandingkan dengan sebelum dilakukan penyerangan. Sebanyak 24 *thread* mengakibatkan *delay* meningkat sebesar 193.13% bila dibandingkan dengan sebelum dilakukan penyerangan.

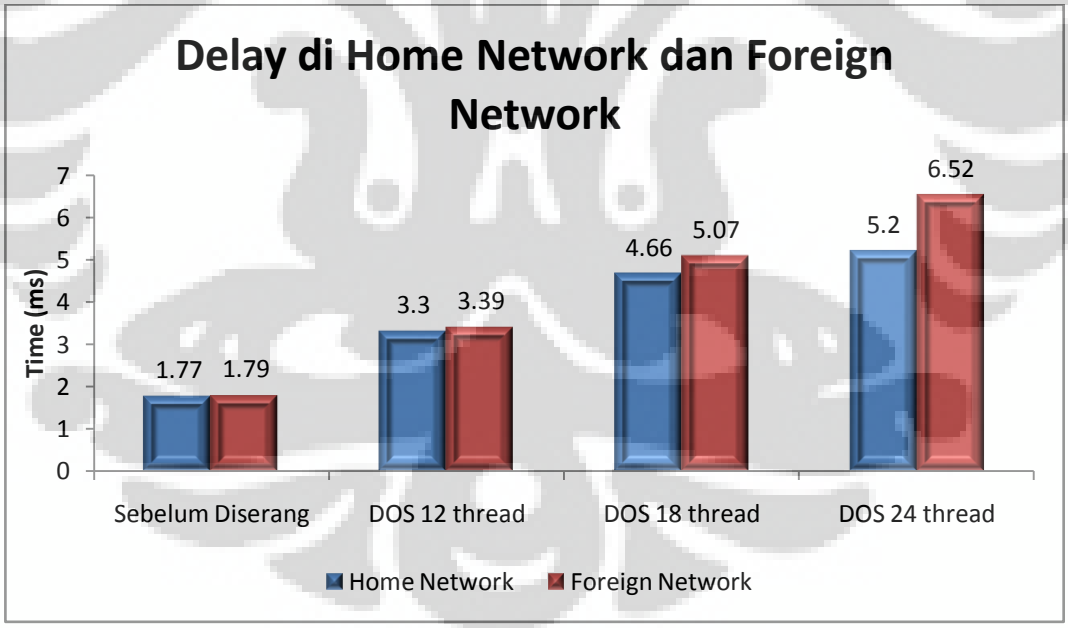
Sementara itu pada *Foreign Network*, *Denial of Service* sebanyak 12 *thread* mengakibatkan *delay* meningkat 89.25% bila dibandingkan dengan sebelum

dilakukan penyerangan. Sebanyak 18 *thread* mengakibatkan *delay* meningkat 182.45% bila dibandingkan dengan sebelum dilakukan penyerangan. Sebanyak 24 *thread* mengakibatkan *delay* meningkat 263.45% bila dibandingkan dengan sebelum dilakukan penyerangan.

c. Perbandingan *Delay* antara *Home Network* dan *Foreign Network*

Analisa perbandingan *delay* di *Home Network* dan *Foreign Network* dibuat dengan menggunakan rata-rata nilai *delay* di *Home Network* dan *Foreign Network*. Berdasarkan rata-rata data yang didapat, nilai *delay* cenderung lebih kecil ketika berada di *Home Network*.

Delay terkecil didapat pada saat skenario tanpa menggunakan serangan di *Home Network* dan *delay* terbesar didapat pada saat skenario dengan menggunakan serangan di *Foreign Network*. Gambar 4.9 memperlihatkan perbandingan grafik rata-rata *delay* untuk masing-masing skenario di *Home Network* dan *Foreign Network*.



Gambar 4.9 *Delay* di *Home Network* dan *Foreign Network*

Dari grafik *delay* tersebut, terdapat 2 kondisi yang dapat dibandingkan, yaitu ketika berada di *Home Network* dan *Foreign Network*. Perbandingan *delay* antara *Home Network* dan *Foreign Network* ini dapat dihitung dengan cara:

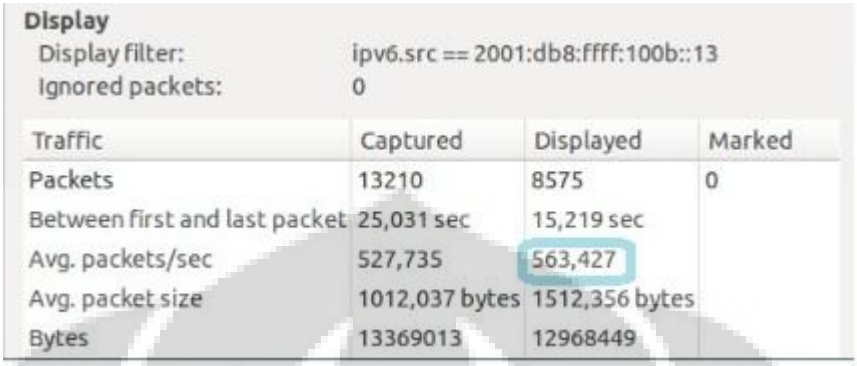
$$\frac{delay\ \ Foreign\ \ Network - delay\ \ Home\ \ Network}{delay\ \ Home\ \ Network} \text{ x } 100\%$$

Pertama yaitu *delay* di *Home Network* lebih kecil daripada di *Foreign Network* sebesar 1.13%. Kedua yaitu *delay* di *Home Network* lebih kecil daripada di *Foreign Network* sebesar 2.72% untuk serangan dengan 12 *thread* pada *Denial of Service*. Ketiga yaitu *delay* di *Home Network* lebih kecil daripada di *Foreign Network* sebesar 8.79% untuk serangan dengan 18 *thread* pada *Denial of Service.* Keempat yaitu *delay* di *Home Network* lebih kecil daripada di *Foreign Network* sebesar 25.38% untuk serangan dengan 24 *thread* pada *Denial of Service.*

Hal ini dapat terjadi karena *Mobile Node* menggunakan metode *Route Optimization* yang menggunakan ipv6 *tunnel*, yang menghubungkan *Mobile Node* langsung ke Correspondont *Node* sehingga data tidak harus melewati *Home Agent* terlebih dahulu. Dengan *tunnel* yang disediakan *Route Optimization* ini, *Mobile Node* seolah-olah tetap berada di *Home Network* walaupun sudah berpindah jaringan ke *Foreign Network*.

4.3.3. Analisa *Throughput*

Throughput adalah sebuah parameter yang merupakan kecepatan *transfer* dari sebuah data dimana besar rata-rata data dapat dikirim setiap detik pada jaringannya. Analisa pada *throughput* dilakukan berdasarkan posisi *Mobile Node* ketika berada di *Home Network* dan *Foreign Network*, baik pada saat sebelum dilakukan penyerangan dan pada saat dilakukan penyerangan dengan Distributed *Denial of Service* dan *Denial of Service*. Pada Summary *Wireshark*, *delay* dapat dilihat pada bagian yang diberi kotak biru pada Gambar 4.10.


Display

Display filter: ipv6.src == 2001:db8:ffff:100b::13
Ignored packets: 0

Traffic	Captured	Displayed	Marked
Packets	13210	8575	0
Between first and last packet	25,031 sec	15,219 sec	
Avg. packets/sec	527,735	563,427	
Avg. packet size	1012,037 bytes	1512,356 bytes	
Bytes	13369013	12968449	

Gambar 4.10 *Throughput* pada Summary *Wireshark*

a. *Throughput* di *Home Network* dan *Foreign Network*

Pertama kali data yang diambil pada analisa *throughput* ini adalah pada saat *Mobile Node* berada di *Home Network* dan *Foreign Network*. *Mobile Node* mengunduh *file* sebanyak 10 kali. Dari hasil pengunduhan tersebut didapatkan data rata-rata *throughput file* tersebut seperti pada Tabel 4.5

Tabel 4.5 Rata-rata *throughput* sebelum diserang

Data ke-	*Throughput* (packet/s)	
	Home Network	*Foreign Network*
1	568.317	563.427
2	570.042	561.968
3	567.6	557.4
4	533.079	551.919
5	569.712	553.917
6	549.48	553.417
7	570.956	556.639
8	569.386	562.231
9	569.629	556.062
10	566.932	554.225
Rata-rata	563.513	557.121

Berdasarkan Tabel 4.5 dapat dilihat perbedaan *throughput* pada saat *Mobile Node* berada di *Home Network* dan *Foreign Network*. Besar data yang diunduh sama, perbedaan *throughput* tidak berbeda jauh walaupun *Mobile Node* sudah berpindah *network*.

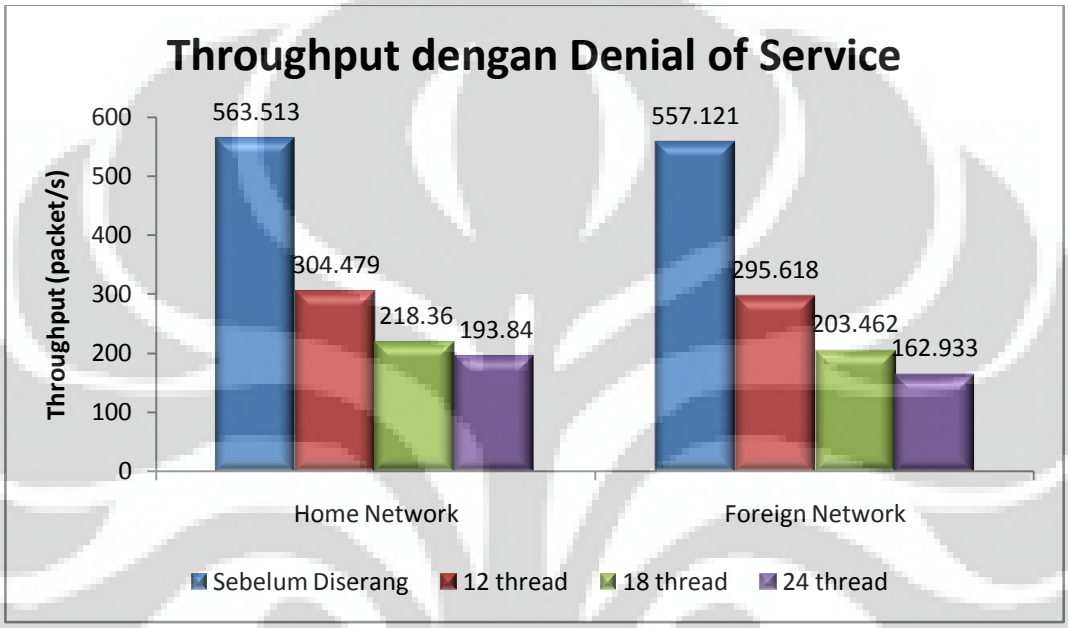
b. *Throughput* dengan Serangan *Denial of Service*

Pengambilan data yang kedua yaitu ketika *Mobile Node* melakukan pengunduhan di *Home Network Foreign Network*. *Mobile Node* mengunduh *file* sebesar 11.75 MB sebanyak 10 kali untuk masing-masing *file* tersebut. Pada saat yang sama dilakukan penyerangan ke FTP *Server* dengan metode *Denial of Service*. Selanjutnya didapatkan rata-rata *throughput* untuk tiap ukuran *file* tersebut seperti pada Tabel 4.8

Tabel 4.6 Rata-rata *throughput* pada *Home Network* dengan Serangan *Denial of Service*

Data ke-	*Throughput* (packet/s)					
	Home Network			*Foreign Network*		
	12	18	24	12	18	24
	(*thread*)					
1	278.799	197.208	184.68	305.083	223.429	196.558
2	338.329	250.142	168.959	281.762	229.6	152.281
3	342.109	205.111	191.808	308.384	223.429	145.405
4	284.115	177.163	212.528	318.528	179.171	149.054
5	311.526	223.85	175.734	286	151.596	219.634
6	267.574	259.77	174.025	315.026	178.496	129.584
7	313.699	232.392	193.253	297.917	215.75	107.348
8	292.93	174.954	224.205	285.723	148.595	114.038
9	318.875	226.714	218.586	305.814	241.313	196.744
10	296.833	236.296	194.626	251.942	243.239	218.684
Rata-rata	304.479	218.360	193.840	295.618	203.462	162.933

Dari Tabel 4.6 dapat dilihat bahwa nilai *throughput* semakin menurun dan berbanding lurus dengan banyaknya *thread* yang digunakan untuk menyerang. Gambar 4.11 menampilkan grafik perbandingan rata-rata *throughput Mobile Node* saat mengunduh *file* di *Home Network* dan *Foreign Network* dengan serangan *Denial of Service*.



Gambar 4.11 *Throughput* dengan *Denial of Service*

Jumlah *thread* yang berbeda berpengaruh secara signifikan karena semakin banyak *thread* yang diberikan, maka *throughput* juga akan semakin bertambah sesuai dengan banyaknya *thread* yang diberikan kepada *Correspondent Node*. Peningkatan *throughput* dapat dihitung dengan cara:

$$\frac{throug\,hput\ \ dengan\ \ thread - throug\,hput\ \ sebelum\ \ diseran\ g}{throug\,hput\ \ sebelum\ \ diserang} \text{ x } 100\ \%$$

Di *Home Network*, *Denial of Service* sebanyak 12 *thread* mengakibatkan *throughput* menurun sebesar 45.96% bila dibandingkan dengan sebelum dilakukan penyerangan. *Denial of Service* sebanyak 18 *thread* mengakibatkan *throughput* menurun sebesar 61.25% bila dibandingkan dengan sebelum dilakukan penyerangan.

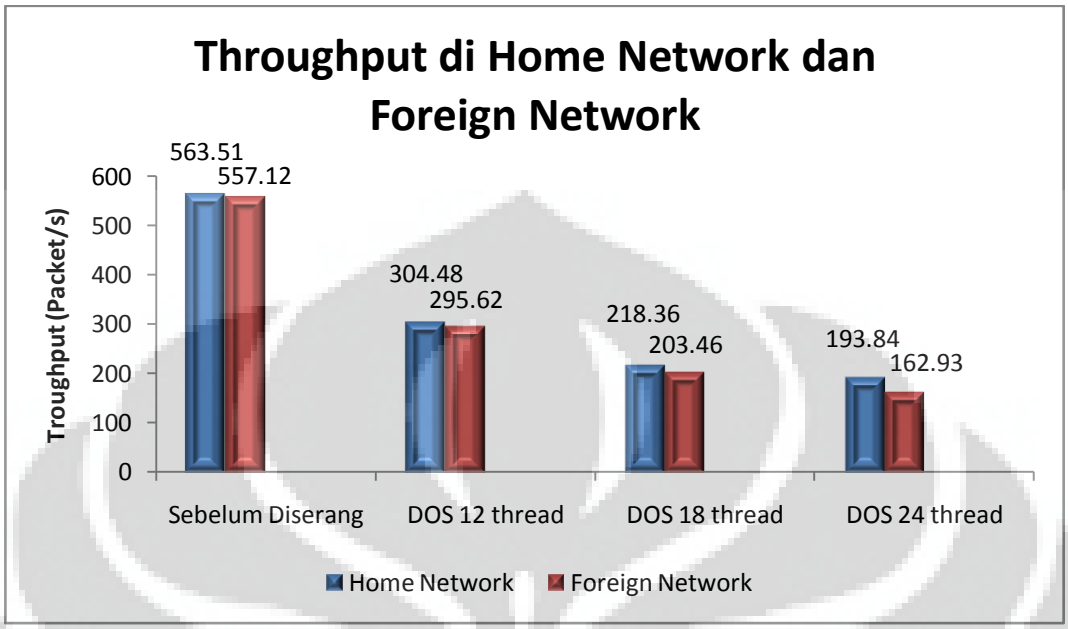
Denial of Service sebanyak 24 *thread* mengakibatkan *throughput* menurun sebesar 65.6% bila dibandingkan dengan sebelum dilakukan penyerangan.

Sementara itu pada *Foreign Network*, *Denial of Service* sebanyak 12 *thread* mengakibatkan *throughput* menurun sebesar 46.94% bila dibandingkan dengan sebelum dilakukan penyerangan. *Denial of Service* sebanyak 18 *thread* mengakibatkan *throughput* menurun sebesar 63.48% bila dibandingkan dengan sebelum dilakukan penyerangan. *Denial of Service* sebanyak 24 *thread* mengakibatkan *throughput* menurun sebesar 70.75% bila dibandingkan dengan sebelum dilakukan penyerangan.

c. Perbandingan *Throughput* diantara Kedua Serangan

Analisa perbandingan *throughput* di *Home Network* dan *Foreign Network* dibuat dengan menggunakan rata-rata nilai *throughput* di *Home Network* dan *Foreign Network*. Berdasarkan rata-rata data yang didapat, nilai *throughput* cenderung lebih besar ketika berada di *Home Network*.

Throughput terbesar didapat pada saat skenario tanpa menggunakan serangan di *Home network* dan *throughput* terkecil didapat pada saat skenario dengan menggunakan serangan di *Foreign Network*. Gambar 4.12 memperlihatkan perbandingan grafik rata-rata *throughput* untuk masing-masing skenario di *Home Network* dan *Foreign Network*.



Gambar 4.12 *Throughput* di *Home Network* dan *Foreign Network*

Dari grafik *throughput* tersebut, terdapat 2 kondisi yang dapat dibandingkan, yaitu ketika berada di *Home Network* dan *Foreign Network*. Perbandingan *throughput* antara *Home Network* dan *Foreign Network* ini dapat dihitung dengan cara:

$$\frac{throug\,hput\ \ Foreign\ \ Network\ \ - throug\,hput\ \ Home\ \ Network}{throug\,hput\ \ Home\ \ Network}\ \text{x}\ 100\%$$

Pertama yaitu *throughput* di *Home Network* lebih tinggi daripada di *Foreign Network* sebesar 1.14%. Kedua yaitu *throughput* di *Home Network* lebih tinggi daripada di *Foreign Network* sebesar 2.99% untuk serangan dengan 12 *thread* pada *Denial of Service*. Ketiga yaitu *throughput* di *Home Network* lebih tinggi daripada di *Foreign Network* sebesar 7.32% untuk serangan dengan 18 *thread* pada *Denial of Service*. Keempat yaitu *throughput* di *Home Network* lebih tinggi daripada di *Foreign Network* sebesar 18.97% untuk serangan dengan 24 *thread* pada *Denial of Service*.

Hal ini dapat terjadi karena *Mobile Node* menggunakan metode *Route Optimization* yang menggunakan ipv6 *tunnel*, yang menghubungkan *Mobile Node* langsung ke Correspondont *Node* sehingga data tidak harus melewati *Home Agent*

terlebih dahulu. Jadi *Mobile Node* seolah-olah tetap berada di *Home Network*, walaupun sudah berpindah jaringan ke *Foreign Network*.

BAB 5
KESIMPULAN

1. Berdasarkan hasil pengukuran, *transfer time* mengalami peningkatan dibandingkan sebelum diserang, pada *Home Network* dengan *Denial of Service* 12 *thread*, 18 *thread* dan 24 *thread* secara berturut - turut yaitu sebesar 76.28%, 159.23% dan 180.2%, sementara pada *Foreign Network* dengan *Denial of Service* 12 *thread*, 18 *thread* dan 24 *thread* secara berturut - turut yaitu sebesar 89.68%, 183.67% dan 266.16%.
2. Berdasarkan hasil pengukuran, *transfer time* di *Home Network* lebih kecil daripada di *Foreign Network*, yaitu sebesar 8.01 % untuk 12 *thread*, sebesar 9.73% untuk 18 *thread* dan 31.05% untuk 24 *thread*.
3. Berdasarkan hasil pengukuran, *delay* mengalami peningkatan dibandingkan sebelum diserang, pada *Home Network* dengan *Denial of Service* 12 *thread*, 18 *thread* dan 24 *thread* secara berturut - turut yaitu sebesar 86.08%, 162.3% dan 193.13%, sementara pada *Foreign Network* dengan *Denial of Service* 12 *thread*, 18 *thread* dan 24 *thread* secara berturut - turut yaitu sebesar 89.25%, 182.45% dan 263.45%.
4. Berdasarkan hasil pengukuran, *delay* di *Home Network* lebih kecil daripada di *Foreign Network*, yaitu sebesar 2.72 % untuk 12 *thread*, sebesar 8.79% untuk 18 *thread* dan 25.38% untuk 24 *thread*.
5. Berdasarkan hasil pengukuran, *throughput* mengalami penurunan dibandingkan sebelum diserang, pada *Home Network* dengan *Denial of Service* 12 *thread*, 18 *thread* dan 24 *thread* secara berturut - turut yaitu sebesar 45.96%, 61.25% dan 65.6%, sementara pada *Foreign Network* dengan *Denial of Service* 12 *thread*, 18 *thread* dan 24 *thread* secara berturut - turut yaitu sebesar 46.94%, 63.48% dan 70.75%.
6. Berdasarkan hasil pengukuran, *throughput* di *Home Network* lebih tinggi daripada di *Foreign Network*, yaitu sebesar 2.99 % untuk 12 *thread*, sebesar 7.32% untuk 18 *thread* dan 18.97% untuk 24 *thread*.

7. Dari hasil keseluruhan pengukuran parameter yang digunakan, dapat disimpulkan bahwa performansi jaringan *Mobile* IPv6 rentan terhadap serangan *Denial of Service*.

DAFTAR ACUAN

[1] Mobile Networking.,"Research Area". [Accessed: 23 November 2012]: http://monet.postech.ac.kr/research.html

[2] BBK.,"Application Layer"., [Accessed: 23 November 2012]: http://penguin.dcs.bbk.ac.uk/academic/*network*s/application-layer/ftp/index.php

[3] Interlink Networks., "RADIUS Server vs. VPN - Link Layer and Network Layer *Security* for *Wireless* Networks"., [Accessed: 23 November 2012]: http://www.interlink*network*s.com/whitepapers/Link_Layer_*Security*.htm

[4] Cisco. "Mobile IP"., [Accessed: 9 Maret 2013]: http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_4-2/*Mobile*_ip.html

[5] NTC Hosting. "What is an FTP *Port*?"., [Accessed: 11 Maret 2013]: http://www.ntchosting.com/ftp/ftp-*port*-connection.html

[6] Wing FTP Software., "Wing FTP Server"., [Accessed: 9 Maret 2013]: http://www.wftpserver.com/

[7] Free and Open InfoSec Education., "Hash Algorithm Collision DoS attack with Xenotix Hash DoS Tester" [Accessed: 18 April 2013]: http://keralacyberforce.in/hash-algorithm-collision-dos-attack-with-xenotix-hash-dos-tester/

[8] L. Osborne, A. Abdel-Hamid, R. Ramadugu. "A Performance Comparison of Mobile IPv6, Hierarchical Mobile IPv6, and Mobile IPv6 Regional Registrations". International Conference on *Wireless* Networks, Communications and Mobile Computing. 2005

[9] A. Moravejosharieh, H. Modares, R. Salleh. "Overview of Mobile IPv6 *Security*". Third International Conference on Intelligent Systems Modelling and Simulation. 2012.