



**UNIVERSITAS INDONESIA**

**APLIKASI SECURE FILE TRANSFER PROTOKOL PADA  
MOBILE IPV6 DAN TUNNELING 6TO4**

**SKRIPSI**

**W A F I R  
0806366472**

**FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
DEPOK  
DESEMBER 2010**



**UNIVERSITAS INDONESIA**

**APLIKASI SECURE FILE TRANSFER PROTOKOL PADA  
MOBILE IPV6 DAN TUNNELING 6TO4**

**SKRIPSI**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar  
Sarjana Teknik**

**W A F I R  
0806366472**

**FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
DEPOK  
DESEMBER2010**

## HALAMAN PERNYATAAN ORISINALITAS

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

**Nama : W A F I R**

**NPM : 0806366472**

**Tanda Tangan :**

**Tanggal :**

## HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama : Wafir

NPM : 0806366472

Program Studi : Teknik Elektro

Judul Skripsi : Aplikasi Secure File Transfer Protokol Pada Mobile IPv6 dan Tunneling 6to4

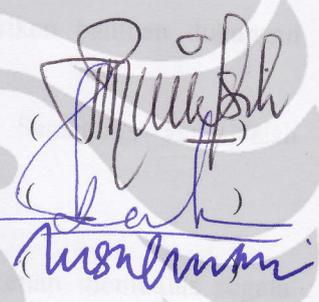
Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia

### DEWAN PENGUJI

Pembimbing : Ir. Endang Sriningsih MT., Si

Penguji : Dr. Ir. Anak Agung Putri Ratna M.Eng

Penguji : Muhammad Salman ST., MIT



Ditetapkan di : Depok

Tanggal : Januari 2011

## UCAPAN TERIMA KASIH

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Jurusan Teknik Elektro pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, saya mengucapkan terima kasih kepada:

- (1) Ir. A Endang Sriningsih MT.,Si., selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan skripsi ini;
- (2) PT. Vieano Trimitra Sejahtera yang telah memberikan bantuan dukungan material dan moral; dan
- (3) SMK Telekomunikasi Tunas Harapan RSBI Kab. Semarang, yang telah menyediakan Lab. ITnya

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu

Depok, 2010

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
SKRIPSI UNTUK KEPENTINGAN AKADEMIS**

---

---

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

Nama : W A F I R  
NPM : 0806366472  
Program Studi : Teknik Elektro  
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 :

**APLIKASI SECURE FILE TRANSFER PROTOKOL PADA  
MOBILE IPV6 DAN TUNNELING 6TO4**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok  
Pada tanggal : Desember 2010  
Yang menyatakan

( W A F I R )

W a f i r  
Departemen Teknik Elektro

Dosen Pembimbing  
Ir. A Endang Sriningsih MT., Si

**APLIKASI *SECURE FILE TRANSFER PROTOCOL* PADA  
*MOBILE IPv6* DAN *TUNNELING 6TO4* UNTUK**

**ABSTRAK**

Tujuan dari penulisan skripsi ini adalah untuk membuat jaringan *Mobile Internet Protokol version 6 (MIPv6)* dan *Tunneling 6to4* untuk melewati Jaringan MIPv6 melewati *Mobile Internet Protocol vesion 4 (MIPv4)*. Dan menerapkan aplikasi *Secure File Transfer Protokol (SFTP)* pada kedua konfigurasi jaringan. Proses pengambilan data menggunakan jaringan local sederhana. Dalam pengujian digunakan sebuah laptop diterapkan sebagai mobile node serta 6 unit PC sebagai router MIPv6 dan tunnel *dual stack*. Pengambilan data dilakukan dengan cara meng-*upload* dan *download* file yang ukurannya berbeda-beda dari mobile node ke home agent dengan perpindahan *access point* yang berpindah – pindah selama *upload-download*. Parameter uji coba yang dibandingkan adalah *transfer time, throughput, dan delay*. Konfigurasi MIPv6 memiliki nilai *transfer time, throughput, dan delay* yang lebih baik dari konfigurasi MIPv6 *tunneling 6to4*. Untuk jaringan yang menggunakan MIPv6 secara presentasi memiliki nilai transfer time rata-rata lebih kecil pada saat upload 12% sampai 29% dan saat download 31% sampai 41% dari konfigurasi jaringan MIPv6 tunneling 6to4. Konfigurasi MIPv6 memiliki nilai throughput lebih besar pada saat upload 12% sampai 36% dan pada saat download 31% sampai 41% dibanding konfigurasi MIPv6 tunneling 6to4. Untuk delay dari konfigurasi MIPv6 lebih kecil pada saat upload 12% sampai 38% dan pada saat download 17% sampai 41% dibanding MIPv6 tunneling 6to4. Perbedaan waktu *hand over* dipengaruhi perangkat *access point*.

**Kata kunci : MIPv6, SFTP, Tunneling, Transfer time, throughput, delay.**

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PERNYATAAN ORISINALITAS .....	ii
HALAMAN PENGESAHAN .....	iii
KATA PENGANTAR .....	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS .....	v
ABSTRAK .....	vi
ABSTRACT .....	vii
DAFTAR ISI .....	viii
DAFTAR GAMBAR .....	xi
DAFTAR TABEL .....	xiii
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Tujuan Penulisan .....	2
1.3 Batasan Masalah .....	2
1.4 Metodologi Penulisan .....	2
1.5 Sistematika Penulisan .....	3
<b>BAB 2 MOBILE IPV6 DAN FILE TRANSFER PROTOKOL (FTP) .....</b>	<b>4</b>
2.1 Mobile Internet Protokol .....	4
2.1.1 Latar Belakang Perkembangan Mobile IP .....	4
2.1.2 Protokol Mobile IPV6 .....	6
2.1.3 Perbandingan Mobile IPV4 dengan Mobile IPV6 .....	9
2.2 Interkoneksi IPV6 ke IPV4 dengan Mekanisme Automatic Tunneling .....	11
2.2.1 Implementasi Automatic Tunneling .....	13
2.3 Secure File Transfer Protokol (SFTP) .....	18
<b>BAB III PERANCANGAN TOPOLOGI JARINGAN .....</b>	<b>19</b>

3.1 Perancangan Topologi Jaringan .....	21
3.2 Konfigurasi Jaringan Mobile IPv6 dengan Debian.....	21
3.2.1 Topologi.....	21
3.2.2 Konfigurasi Home Agent.....	21
3.2.3 Konfigurasi Foreign Agent 1 .....	22
3.2.4 Konfigurasi di Foreign Agent 2 .....	24
3.3 Konfigurasi Mobile Internet Protocol versi 6 (MIPv6) Tunneling 6to4.....	27
3.3.1 Topologi.....	27
3.3.2 Konfigurasi Router IPv4 .....	27
3.3.3 Konfigurasi di Home Agent.....	28
3.3.4 Konfigurasi di Foreign Agent 1 .....	29
3.4 Metode Pengambilan Data .....	32
<b>BAB IV ANALISA DATA .....</b>	<b>34</b>
4.1 Analisa Konfigurasi Jaringan.....	34
4.1.1 Konfigurasi IPv6 .....	34
4.1.2 Konfigurasi Mobile IPv6 dengan Tunneling 6to4 .....	36
4.2 Analisa Performa Jaringan Pada SFTP .....	36
4.2.1 Analisa Transfer Time .....	42
4.2.2 Analisa Troughput .....	44
4.2.3 Analisa <i>Delay</i> .....	47

## **BAB V KESIMPULAN**

..... **50**

## **DAFTAR REFERENSI**

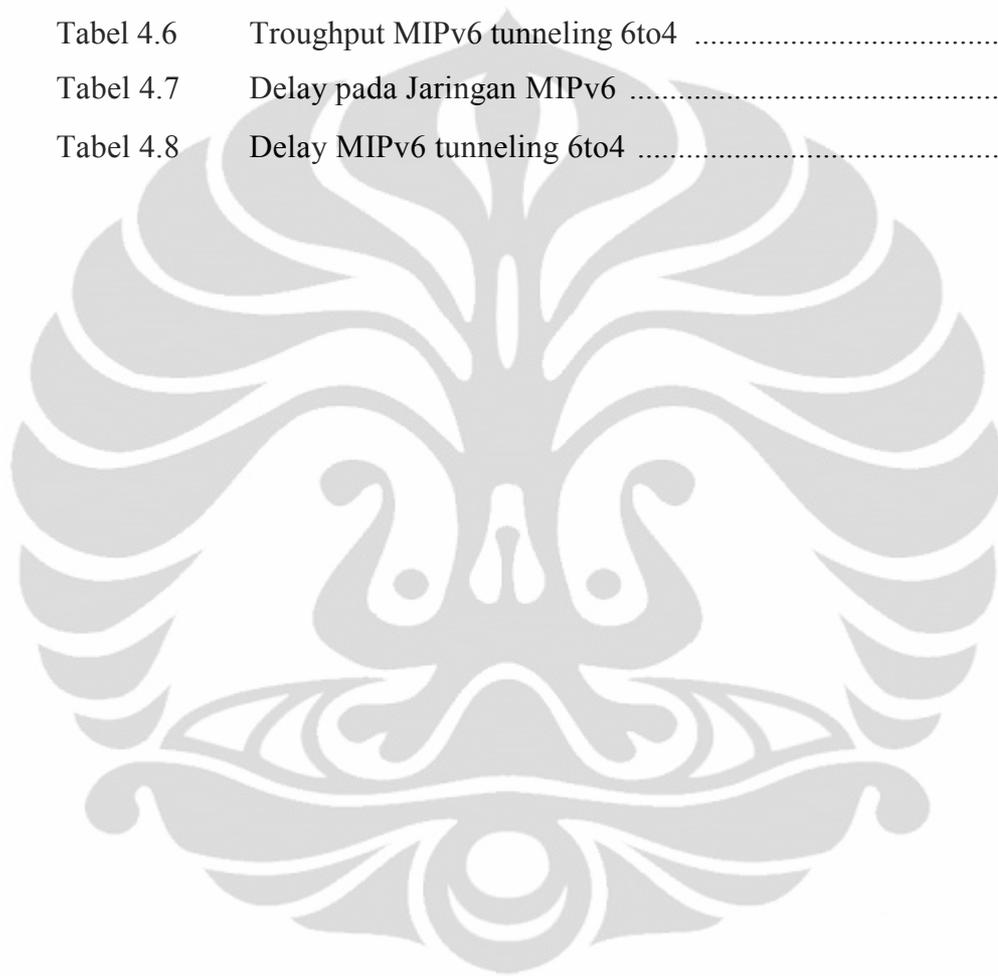
..... **51**

## **DAFTAR LAMPIRAN**

..... **52**

## DAFTAR TABEL

Tabel 4.1	Transfer time MIPv6 .....	34
Tabel 4.2	Transfer time MIPv6 tuneling 6to4 .....	43
Tabel 4.5	Troughput MIPv6 .....	45
Tabel 4.6	Troughput MIPv6 tunneling 6to4 .....	45
Tabel 4.7	Delay pada Jaringan MIPv6 .....	47
Tabel 4.8	Delay MIPv6 tunneling 6to4 .....	47



## DAFTAR GAMBAR

Gambar 2.1	Dari Coresponden Node ke Mobile Node	8
Gambar 2.2	Dari mobile node ke koresponden node	9
Gambar 2.3	Dari koresponden node ke mobile node	10
Gambar 2.4	interkoneksi IPv6 melewati IPv4	16
Gambar 3.1	Topologi jaringan Mobile Internet Protokol versi 6 (MIPv6)	21
Gambar 3.2	Topologi MIPv6 tunneling 6to4	27
Gambar 4.1	Tampilan Tracerouter dari Home Agent	35
Gambar 4.2	Tampilan Tracerouter dari Foreign agent	35
Gambar 4.3	Tampilan aplikasi WinSCP	37
Gambar 4.4	Tampilan file yang terdapat pada MN dan HA	37
Gambar 4.5	Tampilan eksekusi upload/download	38
Gambar 4.6	..... Tampilan donload/upload berlangsung	38
Gambar 4.7	Tampilan pindah access point	39
Gambar 4.8	Tampilan proses koneksi acces point baru	40
Gambar 4.9	Tampilan proses hand over	40
Gambar 4.10	Tampilan proses upload/download setelah perpindahan koneksi dengan Access point baru	41
Gambar 4.11	Pengambilan nilai Transfer time	43
Gambar 4.13	Pengambilan Data Troughput	45
Gambar 4.14	Diagram Perbandingan Throughput	46
Gambar 4.15	Diagram Perbandingan Delay	48

# BAB 1

## PENDAHULUAN

### 1.1 LATAR BELAKANG

Seiring dengan berkembangnya teknologi pada zaman sekarang teknologi komunikasi berbasis *Mobile Internet Protokol* (MIP). Sehingga memungkinkan banyaknya penggunaan *Mobile Internet Protokol version 4* (MIPv4) yaitu teknologi yang digunakan saat ini akan mengalami masalah pengalamatan yang semakin terbatas. Karena keterbatasan jumlah pengalamatan pada MIPv4 ini, sehingga dibutuhkan suatu standart baru pada routing protocol yang mampu mengakomodasi jumlah pengalamatan yang lebih banyak. Berangkat dari masalah ini, *Mobile Internet Protokol version 6* (MIPv6) kemudian dikembangkan dan dijadikan standart baru yang kelak akan mengakomodasi pengalamatan yang lebih banyak.

MIPv6 adalah protocol baru yang didisain untuk menggantikan MIPv4. Alamat MIPv4 pada dasarnya menggunakan metode pengalamatan berbasis 32 bit, yang berarti mampu mengakomodasi jumlah pengalamatan sampai dengan 2 pangkat 32 atau sekitar  $4,294 \times 10$  pangkat 9. Sedangkan IPv6 menggunakan metode pengalamatan berbasis 128 bit, yang berarti mampu mengakomodasi jumlah pengalamatan sampai dengan 2 pangkat 128 atau sekitar  $3,402 \times 10$  pangkat 38. Dengan perbandingan jumlah pengalamatan yang begitu besar inilah, yang mendasari perubahan dari MIPv4 menjadi MIPv6.

MIPv6 merupakan MIP generasi berikutnya atau disebut juga *Internet Protocol Next Generation* (IPng). MIPv6 dirancang sedemikian rupa agar memiliki kinerja yang lebih handal bila dibandingkan dengan IPv4 seperti dalam pengiriman paket, *security*, *authentication* dan QoS (*Quality Of Service*). Selain itu diharapkan MIPv6 juga mampu memberikan fitur-fitur lain yang lebih kompleks yang akan dikembangkan lagi.

Sampai saat ini, secara umum jaringan masih menggunakan IPv4 sehingga implementasi jaringan MIPv6 dilakukan secara bertahap dan diusahakan tidak akan mengganggu jaringan MIPv4 yang sudah ada saat ini. Oleh karena itu, diperlukan suatu mekanisme transisi untuk mengganti penggunaan jaringan

MIPv4 menjadi jaringan MIPv6 secara keseluruhan. Untuk melakukan proses transisi dari MIPv4 ke MIPv6 maka diperlukan suatu metode yang mampu menunjang mekanisme transisi tersebut. Beberapa metode telah diteliti dan diantaranya yaitu metode *Tunneling* dan *Translation*.

Salah satu aplikasi yang akan diimplementasikan sekaligus menjadi uji coba dalam MIPv6 dan MIPv6 *tunneling 6to4* adalah *transfer file*, yang menggunakan *protocol* yang disebut *Secure File Transfer Protocol* (SFTP). Kebutuhan akan proses *upload* dan *download* ke suatu *Home Agent* menjadikan aplikasi ini sering dipergunakan.

## 1.2 TUJUAN SKRIPSI

Tujuan dari skripsi ini adalah sebagai berikut.

1. Merancang sebuah jaringan Mobile Internet Protocol version 6 (MIPv6)
2. Membuat jaringan Tunneling 6to4 untuk melewati MIPv6 dalam jaringan IPv4 yang sudah ada saat ini.
3. Aplikasi *Secure Transfer File Protocol* (SFTP) untuk uji coba *transfer time, throughput dan delay MIPv4 dan MIPv6 tunneling 6to4*.

## 1.3 BATASAN MASALAH

Pada skripsi ini hanya akan membahas pada *Mobile Internet Protokol Internet version 4* (MIPv6) dan MIPv6 dengan *tunneling 6to4*.

Rancangan jaringan sederhana yang akan dibangun adalah jaringan lokal yang menggunakan Router PC dengan system operasi Debian Leny 5.

Aplikasi yang akan diimplementasikan pada jaringan test bed adalah aplikasi WinSCP pada router PC dan *Mobile Node*.

## 1.4 METODOLOGI PENULISAN

1. Studi Literatur

Mengumpulkan dan mempelajari referensi tentang jaringan MIPv6 , tunneling , SFTP, Aplikasi Wireshark.

2. Perancangan Sitem

Pada skripsi ini dirancang system perangkat keras yang diperlukan dapat berjalan lancar dengan perangkat lunak yang diaplikasikan dalam perangkat keras tersebut.

### 3. Pembuatan Sistem

Pembuatan sistem dapat dilakukan setelah semua perangkat keras dan lunak telah terpenuhi. Permasalahan perangkat keras berupa router PC.

### 4. Pengambilan dan analisa data

Setelah dilakukan imlementasi, akan dicatat data-data yang berhubungan dengan *Secure File Transfer Protocol* pada masing-masing jaringan. Dan parameter yang diambil adalah *transfer time*, *throughput*, dan *delay*.

### 5. Penarikan kesimpulan

Penarikan kesimpulan dapat diambil dari topologi masing-masing jaringan yang akan dibuat.

## 1.5 SISTEMATIKA PENULISAN

Dalam penulisan skripsi ini akan disusun secara sistematis yang terdiri atas bagian-bagian yang saling berhubungan sehingga diharapkan akan mudah dipahami dan dapat diambil manfaatnya. Bab satu berisi latar belakang, tujuan skripsi, batasan masalah, dan sistematika penulisan. Bab dua berisi tentang pengenalan tentang *Jaringan Mobile internet Protokol*, tunneling dual stack, dan uraian pendukung jaringan. Bab tiga menjelaskan perancangan *Topologi jaringan* dan konfigurasi jaringan. Bab empat analisa data berisikan tentang pengambilan data dan analisa data masing masing jaringan untuk aplikasi *Secure File Transfer Protocol* (SFTP). Kemudian bab lima sebagai penutup berisikan beberapa kesimpulan dari skripsi yang dilakukan.

## BAB 2

### ***MOBILE IPv6 DAN SECURE FILE TRANSFER PROTOCOL (FTP)***

#### ***2.1 Mobile Internet Protocol***

##### ***2.1.1 Latar Belakang Perkembangan Mobile Internet Protocol***

Semakin pesat perkembangan teknologi komunikasi dan informasi terutama dalam bidang komunikasi *wireless* sehingga kebutuhan akan *mobile* semakin tinggi. Sedangkan untuk setiap perpindahan jaringan terjadi perubahan nomer IP (*internet protocol*). Dengan demikian diperlukan teknologi yang bisa melakukan fungsi untuk tidak merubah alamat IP meskipun berpindah dari suatu jaringan dengan jaringan lainnya. Teknologi yang bisa melakukan fungsi ini adalah *mobile IP*. Dimana dalam teknologi ini ketika sebuah *mobile node* berpindah dari jaringan satu ke lainnya maka tidak mengalami perubahan IP. Dengan kata lain sebuah *mobile node* akan mempunyai alamat yang tetap meskipun selalu berpindah jaringan.

Semakin bertambah *mobile node* yang berbeda pada suatu jaringan computer mengakibatkan kebutuhan akan IP semakin meningkat sehingga untuk memenuhi kebutuhan ini diperlukan adanya alokasi IP yang lebih banyak. Dalam teknologi *Mobile Internet Protocol* (MIP) terdapat dua model : yaitu model IP versi 4 dan *mobile IP* versi 6. *Mobile IP* versi 6 ini mendukung adanya koneksi yang lebih cepat karena didukung adanya teknologi *tunneling*. Yaitu *bidirectional tunnel* dan *route optimization*. Dengan MIP ini diharapkan akan lebih memudahkan dalam pengaturan IP.

Dalam jaringan internet yang menggunakan kabel, ditetapkan bahwa alamat IP mengidentifikasi secara unik titik node yang terhubung pada internet. Karena itu sebuah *node* harus ditempatkan pada jaringan yang diidentifikasi oleh alamat IPnya dalam rangka untuk menerima datagram yang ditunjukkan kepadanya jika tidak, datagram yang ditunjukkan kepada *node* tidak terkirim. Untuk sebuah

*node* yang merubah *point of attachment* tanpa kehilangan kemampuan untuk berkomunikasi, maka salah satu dari mekanisme berikut harus dilakukan:

1. *Node* harus merubah alamat IP nya ketika *node* merubah titik hubungannya ke internet.
2. Route tertentu *node* harus disebarakan ke seluruh penyedia internet

Kedua alternatif ini sering tidak dapat diterima , alternatif pertama tidak mungkin bagi sebuah *node* untuk menjaga sambungan layer transport dan layer yang lebih tinggi ketika *node* merubah lokasinya. Alternatif kedua jelas akan menjadi masalah karena diperlukan mekanisme baru untuk mengakomodasikan *mobilitas node* dalam internet yang memungkinkan *node* merubah alamat IPnya.

Fitur dari mobile IP ini diantaranya yaitu:

1. *Support host* yang berpindah–pindah
2. Tidak ada batasan geografis
3. Tidak ada modifikasi terhadap nomor IP
4. Keamanan jaringan terjamin

Mobile IP yang telah disetujui oleh *Internet Engineering Steering Group (IESG)* pada bulan juni 1996 dan dipublikasikan sebagai *Proposed Standart* pada november 1996, *Proposed Standart* merupakan langkah signifikan pertama pada evolusi sebuah protokol dari draft internet menjadi standart internet penuh.

MIP dibuat oleh IP Routing untuk *wireless/mobile node* yang merupakan kerjasama dengan IETF yang terbentuk pada juni 1992. Dokumen standart dari MIP mengikuti RFC antara lain:

- RFC 2002 yang merupakan protokol dari mobile itu sendiri
- RFC 2003, 2004, dan 1701, yang masing-masing mengidentifikasi tipe tunneling yang digunakan dalam MIP.
- RFC 2005, yang menjelaskan tentang *applicability* dari MIP
- RFC 2006 yang menjelaskan tentang mobile IP *Management information Base (MIB)*. Mobile IP MIB adalah sekumpulan

variabel dalam sebuah *node* yang mengimplementasikan *mobile IP* yang dapat diperiksa atau dikonfigurasi oleh sebuah *manager station* dengan menggunakan *Simple Network Management Protocol versi 2 (SNMPv2)* yang terdapat dalam RFC 1905.

Alamat IP dari sebuah *node* terdiri dari dua bagian yaitu bit alamat yang lebih tinggi menentukan jaringan dimana *node* tersebut terletak dan bit yang lebih rendah menentukan nomor *node* tersebut. *Mobile IP* yang dirancang untuk memungkinkan setiap *mobile node* untuk memiliki dua alamat IP dan mengatur dengan baik proses binding antara dua alamat tersebut. Salah satu dari alamat IP adalah alamat permanen *home address* yang berada pada *home network* dan digunakan untuk komunikasi *endpoint* dan yang lainnya merupakan *temporary care of address* yang menunjukkan lokasi sekarang dimana host tersebut berada. Tujuan utama dari MIP adalah untuk membuat mobilitas dapat semakin mudah untuk dikenali ke level protokol yang lebih tinggi seperti TCP dan untuk meminimalisasi perubahan infrastruktur internet yang ada.

### 2.1.2. Protokol Mobile IPv6

Mobile IPv6 mendefinisikan sebuah protokol IPv6 baru berupa set pesan-pesan dan proses-proses yang digunakan untuk menetapkan hubungan antara node-node yang berdekatan. Protokol tersebut adalah *Neighbor Discovery*.

Proses-proses yang dilakukan oleh *Neighbor Discovery* adalah sebagai berikut :

1. *Router Discovery*, proses dimana sebuah host menelusuri router-router pada sebuah *link*.
2. *Prefix Discovery*, proses dimana host-host menelusuri *prefix-prefix network* untuk link-link local
3. *Parameter Discovery*, proses dimana host-host menelusuri

parameter-parameter operasi tambahan .

4. *Address autoconfiguration*, proses pengkonfigurasi IP address untuk interface-interface secara otomatis.
5. *Neighbor Unreachability detection*, proses dimana sebuah node memastikan bahwa layer IPv6 suatu node tetangga tidak lagi menerima paket-paket.
6. *Neighbor Unreachability detection*, proses dimana sebuah node memastikan bahwa sebuah address yang akan digunakan belum pernah dipakai oleh tetangga.

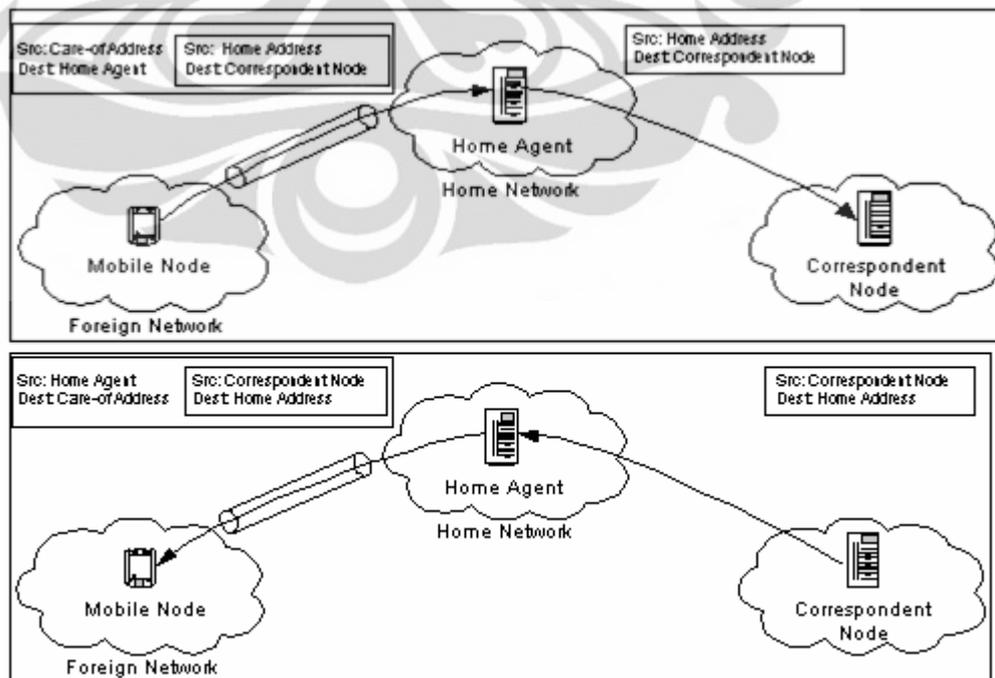
Mobile node selalu diharapkan untuk dialamatkan pada *home addressnya* , meskipun ia berada pada *home linknya* atau jauh dari *home*. *Home address* adalah alamat IP yang diberikan pada mobile node dengan *home subnet* perfixnya pada *home link*. Sementara mobile node berada dalam *home*, paket dialamatkan pada *home addressnya* kemudian dirutekan ke sambungan *mobile node* menggunakan mekanisme routing.

Sementara mobile node menempel pada beberapa *foreign link* yang jauh dari *home*, ia juga dapat dialamatkan pada satu atau lebih *care of address* yang merupakan sebuah alamat IP yang dihubungkan dengan *mobile node* yang mempunyai *subnet perfix* dari sebuah *foreign link* tertentu merupakan. *Mobile node* dapat memperoleh *care of addressnya* melalui mekanisme IPv6 konvensional seperti *stateless* atau *statfull autoconfiguration*. Selama *mobile node* tinggal pada lokasi ini paket dialamatkan pada *care of address* ini untuk kemudian dirutekan ke *mobile node*. *Mobile node* dapat juga menerima paket-paket dari beberapa *care of address*, seperti ketika ia sedang bergerak tetapi masih dapat dicapai pada *link* sebelumnya.

Hubungan antara mobile node, *home address* dan *care of address* dikenal sebagai *corespondent node* . Sementara ketika jauh dari *home*, sebuah *mobile node* meregistrasi *care of address* secara utama dengan *router* pada *home linknya* , permintaan kepada *router* ini berfungsi sebagai *home agent* untuk *mobile node*.

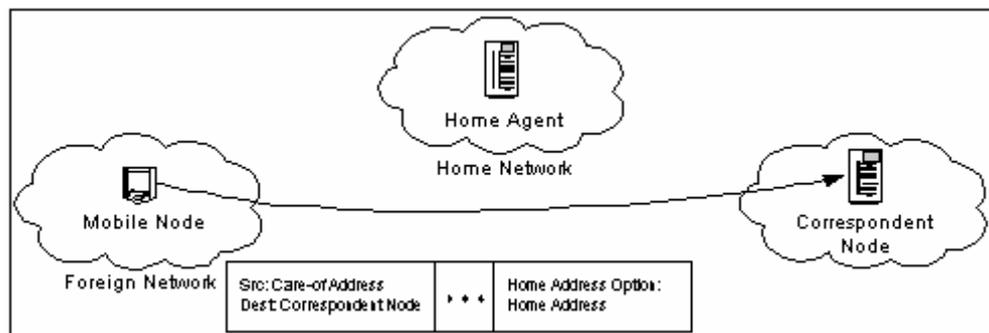
Ada dua mode komunikasi yang mungkin antara *mobile node* dan *correspondent node* yaitu :

1. Mobile node ini membuat *registrasi binding* dengan mengirimkan pesan *binding update* ke *home agent*. *Home agent* akan membalas ke *mobile node* dengan mengembalikan pesan *binding acknowledgement*. Paket-paket dari *correspondent node* dan bahkan tersedia jika *mobile node* tidak meregistrasi bindingnya yang terbaru dengan *correspondent node*. Paket-paket dari correspondent mode dirutekan ke home agent dan kemudian disalurkan dari *mobile node* ke *home agent* (*reverse tunneled*) dan kemudian secara normal dari *home network* ke *correspondent node*. Pada mode ini, *home agent* menggunakan *proxy Neighbor Discovery* untuk menahan beberapa paket IPV6 yang dialamatkan ke *mobile node home address* pada home link. Setiap paket yang ditahan disalurkan ke *mobile node home address* pada home link. Setiap paket yang ditahan disalurkan ke *mobile node primary care of address*. Penyaluran ini menggunakan *enkapsulasi IP6*.



Gambar 2.1 Dari Correspondent Node ke Mobile Node [1]

2. Mode kedua adalah *route optimization*. Mode ini memerlukan dukungan mobile untuk meregistrasi bindingnya pada *corespondent node*. Paket-paket dari *corespondent node* dapat dirutekan secara langsung ke *care of address* dari *mobile node*. Ketika mengirimkan sebuah paket ke beberapa tujuan *corespondent node* mengecek *binding* yang tertahan untuk masukan untuk paket *destination address*. Jika *binding* yang tertahan untuk alamat tujuan ditemukan, node menggunakan sebuah dari tipe dari IPv6 *routing header* yang baru untuk meroutekan paket secara langsung ke *mobile node care of address* membolehkan penggunaan jalur komunikasi terpendek. Ini juga menghilangkan *congestion* pada *mobile node home agent* dan *home link*. Sebagai tambahan dampak dari kemungkinan kegagalan dari *home agent* atau *network* pada jalur dapat dikurangi. Ketika peroutingan paket secara langsung ke *mobile node*, *corespondent node* menyesuaikan *destination address* pada IPv6 *header* ke *node care of address* dari *mobile node*. Tipe *routing IPv6v6 header* yang baru juga ditambahkan ke paket untuk dibawa ke *home address* yang ditentukan, *mobile node* menyesuaikan *source address* dalam IPV6 paket *header* ke *care of addressnya* yang baru. *mobile node* menambahkan pilihan tujuan IPv6 *home address* yang baru untuk membawanya ke *home address*. Pencantuman *home address* pada paket-paket ini membuat penggunaan *care of address transparan* diatas *network layer*.



Gambar 2.2 Dari mobile node ke koresponden node [1]



Gambar 2.3 Dari koresponden node ke mobile node [1]

### 2.1.3 Perbandingan Mobile IPv4 dengan Mobile IPv6

Ada perbedaan utama antara mobile IPv4 dengan mobile IPv6

1. Pada mobile IPv6 tidak ada keharusan untuk menggunakan router khusus sebagai *foreign agent* seperti di mobile IPv4. Mobile IPv6 beroperasi di beberapa lokasi tanpa kebutuhan khusus dari router local.
2. Mobile IPv6 mendukung untuk optimasi rute yang menjadi bagian dasar protokol, daripada perluasan yang standar.
3. Optimasi *route mobile* IPv6 dapat beroperasi secara aman bahkan tanpa *pre-anggered security association*. Ini diharapkan bahwa *optimasi router* tersebut dapat dilakukan skala global antara seluruh *mobile node* dan *corresponden node*.
4. Kebanyakan paket dikirimkan ke *mobile node* sementara jauh dari *home* dalam mobile IPv6 dikirim menggunakan IPv6 *routing header* daripada *enkapsulasi IP*, mengurangi apa yang dikerjakan dalam mobile IPv4.
5. Mobile IPv6 dipisahkan dari beberapa bagian *link layer*, sebagaimana digunakan pada *neighbor Discovery*. Ini juga meningkatkan kekuatan dari protokol.
6. Penggunaan *enkapsulasi IPv6* memindahkan kebutuhan dalam mobile IPv6 untuk mengatur *tunnel soft state*.
7. Mekanisme penemuan *home agent address dinamis* dalam mobile IPv6 mengembalikan balasan tunggal ke mobile node.

Pendekatan *directed broadcast* digunakan dalam IP V4 untuk mengembalikan balasan yang terpisahke setiap home.

## 2.2 *Interkoneksi IPv6 ke IPv4 dengan Mekanisme Automatic Tunneling*

IPv6 mempunyai format alamat dan *header* yang berbeda dengan IPv4 sehingga tidak bisa melakukan *interkoneksi* dengan IPv4 secara langsung karena itu diperlukan suatu mekanisme transmisi IPv6 agar paket IPv6 dapat dilewatkan pada jaringan IPv4 yang telah ada atau sebaliknya. Salah satu metode transisi adalah metode *tunneling*.

Panjang alamat IPv4 sebesar 32 bit , alamat IPv4 pada dasarnya terdiri dari dua bagian utama , yaitu identitas jaringan (*network ID*) dan identitas komputer (*host ID*) . *Netwok ID* menyatakan identitas dari jaringan dimana komputer tersebut berada, sementara *host ID* menyatakan identitas jaringan dimana komputer itu sendiri.

Pembagian kelas IPv4

Kelas A = 00000000 s/d 01111111 = 0 s/d 127

Kelas B = 10000000 s/d 10111111 = 128 s/d 191

Kelas C = 11000000 s/d 11011111 = 192 s/d 223

Kelas D = 11100000 s/d 11101111 = 224 s/d 239

Kelas E = 11110000 s/d 11111111 = 240 s/d 255

Penulisan Alamat IPv4 *Tunneling* : 202.149.240.66 dengan menggunakan contoh ini , jika administrator men-setup jaringan dengan semua komputer yang memiliki bagian nilai yang sama. 202.149.240.xxx angka yang bergaris bawah merupakan network ID-nya, sedangkan xxx adalah host ID-nya. Kelas pada IPV4 mewakili sebuah group alamat yang dapat dikenali *software* sebagai suatu jaringan fisik. Untuk mengetahui suatu alamat IPV4 itu termasuk kelas apa, kita harus merubah alamat IPV4 tersebut ke dalam nilai biner, contoh : 10.149.240.66

00001010.10010101.11110000.10000010

Dengan memperhatikan 3 nilai biner pertama kita bisa mengatakan alamat IPv4 diatas adalah kelas A.

Alamat IPV6 yang mempunyai panjang 128 bit dalam hexadesimal

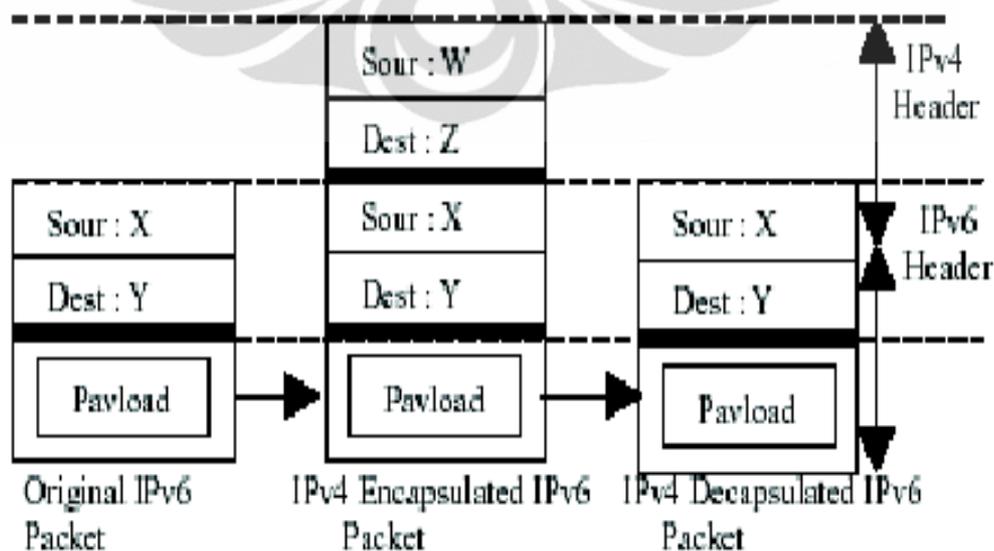
tentunya sulit dihafalkan karena itu alamat numerik jarang digunakan lebih mudah DSN memegang peranan penting. Alamat IPv6 sendiri terbagi atas beberapa macam berdasarkan RFC 3513 :

- *unspecified* dengan notasi `::/128`
- *Loopback* dengan notasi `::1/128`
- *Multicast* dengan notasi `ff00::/8`
- *Link local unicast* dengan notasi `FE80::/8`
- *Site local unicast* dengan notasi `FECO::/8`
- *Global unicast*

Alamat yang akan digunakan untuk berkomunikasi dengan internet adalah alamat global unicast. Pembagian alokasi alamat global berdasarkan registrasi RFC 2928 :

- IANA 2001:000::/29 sampai 2001:01F8::/29
- APNIC 2001:200::/29 sampai 2001:03F8::/29
- ARIN 2001:400::/29 sampai 2001:05F8::/29
- RIPE NCC 2001:400::/29 sampai 2001:05F8::/29

Tunneling protokol merupakan mekanisme proses enkapsulasi atau network protokol yang disebut *payload* protokol ke dalam *delivery* protokol yang berbeda. Tunneling IPv6 over IPv4 merupakan suatu proses *enkapsulasi* paket IPv6 dengan header IPv4 sehingga paket IPv6 dapat dikirim melalui jaringan IPv4 Struktur tunneling IPv6 pada IPv4 ditunjukkan gambar 2.4.



Gambar 2.4 Enkapsulasi paket pada proses tunneling [2]

### 2.2.1 Implementasi Automatic Tunneling

Untuk membangun sebuah sistem mekanisme transisi automatic tunneling diperlukan beberapa langkah :

1. Desain arsitektur sistem automatic tunneling
2. Implementasi 2 *automatic tunneling gateway / Dualstack*
3. Pengaturan routing IPV6 pada *gateway* dan *client tunnel*
4. Pengujian mekanisme *automatic tunneling*

Sebagai contoh implementasi mekanisme tunneling automatic

```
| pinguin1 |-----| CNC 1 |-----| Router / Internet IPv4  
|-----| CNC 2 |-----| pinguin2 |
```

Misalkan dengan konfigurasi seperti contoh diatas dengan kondisi sebagai berikut :

- CNC 1 adalah Personal Computer (PC) *dualstack* dengan  
alokasi *perfix* IPV4 adalah :198.xxx.xxx.x ::/48  
alokasi *perfix* IPV6 adalah :2002:xxxx:xxxx::1/48  
alokasi alamat IPV6 adalah :2002:xxxx:xxxx::1/48
- Pinguin 1 adalah *host* IPV6 dengan  
alokasi IPV6:2002:xxxx:xxxx::2/48
- CNC 2 adalah PC *dualstack* dengan  
alokasi IPV4 adalah:190:xx:xx:x  
alokasi *perfix* IPV6 adalah:2002:xxxx:xxxx::/48  
alokasi alamat IPV6 adalah:2002:xxxx:xxxx::1/48
- Pinguin2 adalah *host* IPV6 dengan  
alokasi alamat IPV6 : 2002:xxxx:xxxx::2/48

**Implementasi *gateway tunneling* pada CNC 1 (*Dualstack*)**  
untuk *host* CNC 1 harus dialokasikan dengan menggunakan alamat

IPV6 global supaya dapat diroutekan pada jaringan IPV6 dan internet. Format alamat IPV6 untuk *gateway tunnel* adalah sebagai berikut :

2002 = prefix global

xxxx:xxxx = alamat IPv4 dalam hexa (198.xxx.xxx.x =  
xxxx:xxxx)

Supaya operating system kita mendukung IPV6 kita perlu instalasi modul IPv6 adapun perintah untuk aktivasi modulnya adalah : `#insmod ipv6`

Misalkan yang terhubung ke jaringan IPV4 adalah interface eth0 dan yang terhubung ke *client tunnel* adalah eth1. Setelah itu kita perlu konfigurasi alamat IPV4 yaitu dengan perintah :

```
#ifconfig eth0 198.xxx.xxx.x netmask 255.255.255.0 up
```

Untuk pengecekan *interface* kita gunakan perintah :

```
#ifconfig eth0
```

Langkah selanjutnya adalah mengaktifkan *interface* untuk *tunneling* yang digunakan untuk membangun jembatan menembus jaringan IPV4. Aktivasinya dengan menggunakan perintah :

```
#ifconfig sit0 up
```

Dengan implementasi *gateway tunneling*nya adalah dengan format sebagai berikut : `::address IPV4 host /prefix`

Dalam implementasi *gateway tunnel address* IPV4 pada eth0 yang digunakan adalah 198.xxx.xxx.x maka format address yang muncul pada *interface* sit0 dan CNC 1 adalah `::198.xxx.xxx.x` dengan prefix 96. Biasanya alamat ini akan terkonfigurasi secara otomatis saat kita aktifasi sit0. Untuk mengkonfigurasinya dapat dilakukan dengan perintah :

```
#ifconfig sit0 add ::198.xxx.xxx.x/196 up
```

Utau untuk menghapus digunakan

```
#ifconfig sit0 del::198.xxx.xxx.x/96
```

Untuk mengecek sit0 menggunakan perintah

```
#ifconfig sit0
```

Untuk selanjutnya mengkonfigurasi alamat pada interface eth1. Alamat IPV6 untuk eth1 harus menggunakan prefix ulang telah dihitung yaitu 2002:xxxx:xxxx::1/48. Perintah yang digunakan :

```
#config eth1 add 2002:xxxx:xxxx::1/48 up
```

Untuk mengecek konfigurasinya menggunakan

```
#ifconfig eth1
```

Langkah yang terakhir yang dikonfigurasi *gateway tunnel* CNC 1 ini adalah dengan mengkonfigurasi entri tabel routingnya yaitu semua alamat yang bertujuan ke CNC 2 atau dengan prefix 2002:xxxx:xxxx::/48 dilewatkan ke *interface tunnel* sit0 , dengan perintah

```
#route-A inet6 add 2002:xxxx:xxxx::/48 gw
```

```
::198.xxx.xxx.x dev sit0
```

supaya *gateway* dapat memforward paket IPV6 maupun IPV4 maka diperlukan aktivasi *IP forward* pada sistem operasi dengan perintah

```
#echo"1">/proc/sys/net/ipv4/conf/all/forwarding
```

```
#echo"1">/proc/sys/net/ipv6/conf/all/forwarding
```

### **Implementasi Gateway Tunneling pada CNC 2**

Gateway Tunnel CNC 2 mempunyai prinsip konfigurasi yang sama. Untuk alamat IPv6 yang dipakai 2002 : prefix global

xxxx:xxxx : Alamat IPv4 CNC 2 dalam hexa (190.xx.xx.x =  
xxxx:xxxx )

Supaya operating system kita mendukung IPv6 kita perlu instalasi modul IPv6 adapun perintah untuk aktivasi modulnya adalah:

```
# insmod ipv6
```

Misalkan yang terhubung ke jaringan IPv4 adalah interface eth0 dan yang terhubung ke *client tunnel* adalah eth1 .Setelah itu kita perlu konfigurasi alamat IPv4 yaitu dengan perintah :

```
# ifconfig eth0 190.xx.xx.x netmask 255.255.255.0 up
```

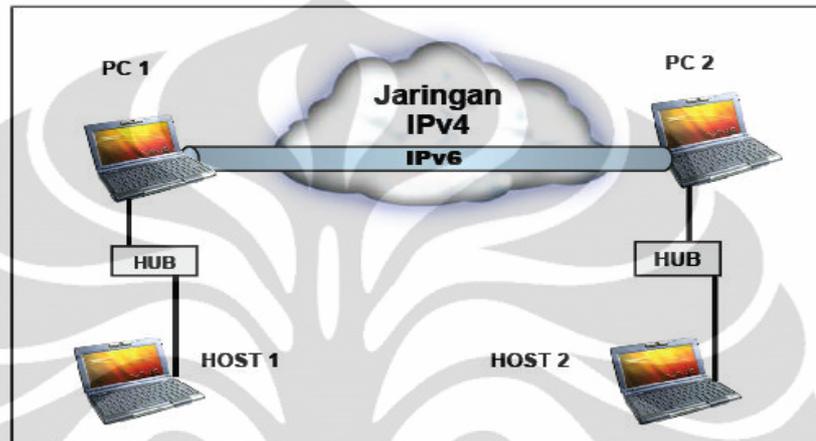
Seperti halnya konfigurasi pada *gateway tunnel* CNC 1

maka kita perlu aktivasi *interface tunnel* sit0 .

```
#ifconfig sit0 up
```

Apabila inet6 address sudah terkonfigurasi alamat IPv6 *compatibel* IPv4 maka kita tidak perlu melakukan penambahan alamat pada interface sit0.

```
#ifconfig sit0 up
```



Gambar 2.5 interkoneksi IPv6 melewati IPv4 [2]

Sebuah perusahaan memiliki topologi jaringan seperti gambar diatas dengan kondisi sebagai berikut :

- PC 1 adalah PC *dualstack* dengan alokasi IPv4 adalah 202.120.120.1  
Alokasi *prefix* IPv6 adalah : 2002:ca78:7801::/48  
Alokasi alamat IPv6 adalah : 2002:ca78:7801::1/48
- HOST 1 adalah *host* IPv6 dengan alokasi IPv6 2002:ca78:7801::2/48
- PC 2 adalah PC *dualstack* dengan alokasi IPv4 adalah 202.81.81.2  
Alokasi *prefix* IPv6 adalah : 2002:ca51:5102::/48  
Alokasi alamat IPv6 adalah : 2002:ca51:5102::1/48
- HOST 2 adalah *host* IPv6 dengan alokasi alamat IPv6  
2002:ca51:5102::2/48
- PC 1 & PC 2 terhubung ke jaringan IPv4 menggunakan interface eth0.
- PC 1 terhubung ke HOST 1 dan PC 2 terhubung ke HOST 2 menggunakan *interface* eth1.

Pada PC 1 alamat IPv4 nya harus dialokasikan dengan menggunakan alamat IPv6

global supaya dapat di rutekan pada jaringan IPv6, pengalokasiannya adalah sebagai berikut :

Prefix Global :	Alamat IPv4 dalam hexadesimal
2002:ca78:7801::/48	

Ket: 2002 = Prefix global

ca78:7801 = Alamat IPv4 PC 1 dalam hexa ( 202.120.120.1 = ca78:7801 )

```
# ifconfig eth0 202.120.120.1 netmask 255.255.255.0 up {
Pengkonfigurasi alamat IPv4 pada PC 1 }
```

```
# ifconfig sit0 up { Mengaktifkan interface untuk tunneling yang
digunakan untuk membangun jembatan menembus jaringan IPv4 }
```

Alamat IPv4 pada eth0 yang digunakan adalah 202.120.120.1 maka format alamat yang dimunculkan pada *interface tunneling* ( sit0 ) PC 1 adalah ::202.120.120.1 dengan prefix 96.

Alamat IPv6 pada eth1 yang digunakan adalah 2002:ca78:7801::1/48

```
# ifconfig eth1 add 2002:ca78:7801::1/48 up
{ Pengkonfigurasi alamat IPv6 pada interface eth1 ( HOST 1 ) }
```

```
# route -A inet6 add 2002:ca51:5102::/48 gw ::202.120.120.1 dev sit0
{Pengkonfigurasi entry table routing / semua alamat yang bertujuan ke
PC 2 akan dilewatkan ke interface tunneling ( sit0 ) }
```

*HOST* 1 berada dibawah layanan PC 1, maka harus dibuat alamat *prefix* dari *HOST* 1 ini sama dengan alamat *prefix* dari PC 1. Alamat Prefixnya yaitu 2002:ca78:7801::/48, maka dapat dialokasikan alamat untuk *HOST*1 yaitu : 2002:ca78:7801::2/48. Semua paket yang bertujuan ke *prefix* PC 2 harus dilewatkan ke eth0 dengan perintah konfigurasi sebagai berikut :

```
#ifconfig eth0 add 2002:ca78:7801::2/48 up
```

```
#route -A inet6 add 2002:ca51:5102::/48 gw 2002:ca78:7801::1 dev eth0
```

Sama halnya juga dengan *HOST* 2 berada dibawah layanan PC 2, maka harus dibuat alamat *prefix* dari *HOST* 2 ini sama dengan alamat *prefix*

dari PC 2. Alamat Prefixnya yaitu 2002:ca51:5102::/48, maka dapat dialokasikan alamat untuk *HOST* 2 yaitu : 2002:ca51:5102::2/48. Semua paket yang bertujuan ke *prefix* PC 2 harus dilewatkan ke eth0 dengan perintah konfigurasi sebagai berikut:

```
#ifconfig eth0 add 2002:ca51:5101::2/48 up
#route -A inet6 add 2002:ca78:7801::/48 gw 2002:ca51:5102::1 dev eth0
```

### 2.3 *Secure File Transfer Protokol (SFTP)*

*Secure Shell* atau SSH adalah protokol jaringan yang memungkinkan pertukaran data melalui saluran aman antara dua perangkat jaringan. Terutama banyak digunakan pada sistem berbasis Linux dan Window untuk mengakses akun shell, SSH dirancang sebagai pengganti Telnet dan *shell remote* tak aman lainnya, yang mengirim informasi, terutama kata sandi, dalam bentuk teks sederhana yang membuatnya mudah untuk dicegat. Enkripsi yang digunakan oleh SSH menyediakan kerahasiaan dan integritas data melalui jaringan yang tidak aman seperti Internet. standar TCP port 22 telah ditetapkan untuk menghubungi server SSH.

Dalam semua versi SSH, penting untuk memverifikasi kunci publik sebelum menerimanya secara valid. Menerima seorang kunci publik *attacker* sebagai kunci publik yang valid memiliki efek membuka password yang ditransmisikan dan memungkinkan serangan *man in-the-middle*. Dalam [komputasi](#), *SSH File Transfer Protokol* (kadang-kadang disebut *Secure File Transfer Protokol* atau SFTP) adalah [protokol jaringan](#) yang menyediakan [akses file](#), [transfer file](#), dan [file manajemen](#) fungsionalitas atas setiap diandalkan [data stream](#). Hal ini dirancang oleh [Internet Engineering Task Force](#) (IETF) sebagai perluasan dari [Secure Shell protocol](#) (SSH) versi 2.0 untuk menyediakan kemampuan transfer file aman. IETF dari [Internet Draft](#) menyatakan bahwa meskipun protokol ini dijelaskan dalam konteks protokol SSH2, protokol ini bersifat umum dan independen . Ini bisa digunakan dalam beberapa aplikasi yang berbeda, seperti transfer file aman selama [Transport Layer Security](#) (TLS) dan transfer informasi manajemen [VPN](#) aplikasi.

## BAB III

### PERANCANGAN TOPOLOGI JARINGAN

#### 3.1 Perancangan Topologi Jaringan

Pada perancangan topologi jaringan *test-bed* terdiri dari 2 tipe jaringan yang diklasifikasikan menurut teknik konfigurasi pengalamatannya, yaitu jaringan Mobile IPv6, dan jaringan Mobile IPv6 *tunneling 6to4*.

Pada kedua tipe jaringan tersebut akan dilakukan pengamatan mengenai kualitas *Security File Transfer Protokol* masing-masing tipe jaringan (SFTP) dari masing-masing tipe jaringan dan memperoleh informasi mengenai kelebihan dan kekurangan dari jaringan tersebut.

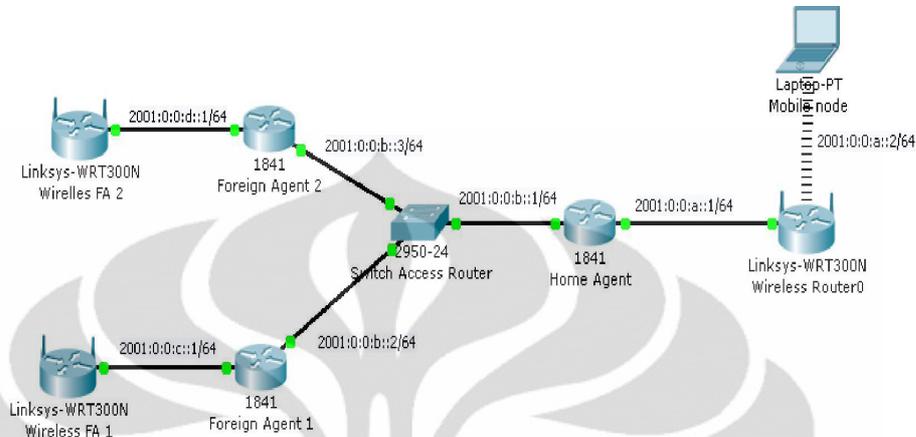
Sebagai perangkat pendukung utama dalam test bed akan digunakan PC yang bertindak sebagai router IPv4 – *Tunneling Dual Stack* sebagai solusi dari permasalahan keterbatasan router yang tidak dimiliki, karena keterbatasan biaya . Berikut ini spesifikasi perangkat keras yang diperlukan dalam jaringan test-bed :

1. *Home Agent* (PC Intel Atom)
  - ⊗ Sistem operasi *Linux Debian Lenny 5* aplikasi WinSCP
  - ⊗ Intel Centrino CPU 2.00 GHz
  - ⊗ 1 GByte RAM
  - ⊗ Ethernet 10/100 Mbp
2. *Mobile Node* (*laptop* Axioo Neon M740SUN)
  - ⊗ Sistem operasi *Windows 7* aplikasi WinSCP dan WireShark
  - ⊗ Intel Core 2 Duo CPU 2.00 GHz
  - ⊗ 2 GByte RAM
  - ⊗ Ethernet 10/100 Mbps1.
3. *Foreign Agent 1* (PC Intel Atom)
  - ⊗ Sistem operasi *Linux Debian Lenny 5* aplikasi WinSCP
  - ⊗ Intel Centrino CPU 2.00 GHz
  - ⊗ 1 GByte RAM
  - ⊗ Ethernet 10/100 Mbps
4. *Foreign Agent 2* (PC Intel Atom)
  - ⊗ Sistem operasi *Linux Debian Lenny 5* aplikasi WinSCP

- ⊗ Intel Centrino CPU 2.00 GHz
  - ⊗ 1 GByte RAM
  - ⊗ Ethernet 10/100 Mbps
4. *Router IPv4 (PC Intel Atom)*
- ⊗ Sistem operasi *Debian lenny 5*
  - ⊗ Pentium 4 CPU 2.00 GHz
  - ⊗ 1 GByte RAM
  - ⊗ Ethernet 10/100 Mbps4.
5. *Dual Stack 1 (PC Intel Atom)*
- ⊗ Sistem operasi *Linux Debian Lenny 5 Aplikasi WinSCP*
  - ⊗ Intel Centrino CPU 2.00 GHz
  - ⊗ 1 GByte RAM
  - ⊗ Ethernet 10/100 Mbps
5. *Dual Stack 1 (PC Intel Atom)*
- ⊗ Sistem operasi *Linux Debian Lenny 5 aplikasi WinSCP*
  - ⊗ Intel Centrino CPU 2.00 GHz
  - ⊗ 1 GByte RAM
  - ⊗ Ethernet 10/100 Mbps
6. Enam unit LAN Card (TP-Link)
- ⊗ Supports 10/100 Mbps
  - ⊗ *Integrated Fast Ethernet MAC*
7. Enam unit *Wireless router* (Linksys)
- ⊗ Supports 802.11b/g/n *standars*
  - ⊗ Kecepatan hingga 300 Mbps
  - ⊗ Maximum 100 m indoor dan 300 m outdoor
- 8 . Swich (TP-Link)
9. Dua Kabel UTP Cross
10. Delapan Kabel UTP Straigh

## 3.2 Konfigurasi Jaringan Mobile IPv6 dengan Debian

### 3.2.1. Topologi



Gambar 3.1 Topologi jaringan *Mobile Internet Protokol version 6* (MIPv6)

Pada gambar 3.1 merupakan gambaran topologi jaringan Mobile IPv6 , eth0 Home Agent (HA) dihubungkan dengan switch, sementara eth1 HA dihubungkan dengan accespoint HA (wireless router 0) . Eth0 Foreign Agent 1 (FA1) dihubungkan dengan switch, sementara eth1 dihubungkan dengan accespoint FA1 (wireless FA1). Eth0 Foreign agent 2 (FA2) dihubungkan dengan switch, sementara eth1 FA2 dihubungkan accespoint FA2 (wireless FA2).

### 3.2.2. Konfigurasi Home Agent

1. Login sebagai root
2. Lalu ketikkan `modprobe ipv6`

***HomeAgent:~# modprobe ipv6***

3. Setelah itu ditambahkan IP address sesuai dengan topologi

***HomeAgent:~# nano /etc/network/interfaces***

4. Tambahkan baris seperti berikut

***auto lo***

***iface lo inet loopback***

*auto eth0*

*iface eth0 inet6 static*

*address 2001:0:0:b::1*

*netmask 64*

*auto eth1*

*iface eth1 inet6 static*

*address 2001:0:0:a::1*

*netmask 64*

( lalu keluar dengan menekan tombol ctrl + x jawab y lalu enter ) 5.

Setelah itu tambahkan proses networknya dengan perintah

*HomeAgent:~# /etc/init.d/networking restart*

*echo 1 > /proc/sys/net/ipv6/conf/all/forwarding*

*route -A inet6 add 2001:0:0:c::/64 gw 2001:0:0:b::2 dev eth0*

*route -A inet6 add 2001:0:0:d::/64 gw 2001:0:0:b::3 dev eth0*

### **3.2.3 Konfigurasi Foreign Agent 1**

1. Login sebagai root

2. Mount Paket dari CD/DVD

*apt-cdrom add*

3. Install Paket ssh-server

*apt-get install ssh-server*

4. Berikan IPv4 terlebih dahulu untuk menambah packet yang tidak ada di CD/DVD

*ifconfig eth0 192.168.0.1 netmask 255.255.255.0 up*

5. Koneksikan PC client ke Foreign Agent lalu berikan ip pada LAN

*ip address 192.168.0.2*

Tambahan packet yang belum ada di CD/DVD menggunakan WinScp

*radvd*

6. reboot PC Foreign Agent

Setelah proses *reboot* selesai lakukan konfigurasi

1. Login sebagai root

2. Lalu ketikkan *modprobe ipv6*

*ForeignAgent:~# modprobe ipv6*

3. Setelah itu tambahkan IP address sesuai dengan topologi

*ForeignAgent:~# nano /etc/network/interfaces*

4. Tambahkan baris seperti berikut

*auto lo*

*iface lo inet loopback*

*auto eth0*

*iface eth0 inet6 static*

*address 2001:0:0:b::2*

*netmask 64*

*gateway 2001:0:0:b::1*

*auto eth1*

*iface eth1 inet6 static*

*address 2001:0:0:c::1*

*netmask 64*

( lalu keluar dengan menekan tombol *ctrl + x* jawab *y* lalu *enter* )

5. Setelah itu tambahkan proses networknya dengan perintah

***ForeignAgent:~# /etc/init.d/networking restart***

***echo 1 > /proc/sys/net/ipv6/conf/all/forwarding***

***route -A inet6 add 2001:0:0:a::/64 gw 2001:0:0:b::1 dev eth0***

***route -A inet6 add 2001:0:0:d::/64 gw 2001:0:0:b::3 dev eth0***

6. Kemudian lakukan instalasi paket *radvd* yang telah ditambahkan tadi

***ForeignAgent:~# dpkg -i radvd.deb***

7. Lalu buat konfigurasi *radvd.conf* di */etc* dengan cara

***ForeignAgent:~# nano /etc/radvd.conf***

8. Lalu tambahkan baris berikut

***interface eth1***

***{***

***AdvSendAdvert on;***

***prefix 2001:0:0:c::/64***

***{***

***};***

***};***

9. Lalu restart *radvd* dengan perintah

***ForeignAgent:~# /etc/init.d/radvd restart***

### **3.2.4 Konfigurasi di Foreign Agent 2**

1. Login sebagai root

2. Mount Paket dari CD/DVD

***apt-cdrom add***

3. Install Paket *ssh-server*

***apt-get install ssh-server***

4. Berikan IPv4 terlebih dahulu untuk menambah packet yang tidak ada di CD/DVD

```
ifconfig eth0 192.168.0.1 netmask 255.255.255.0 up
```

5. Koneksikan PC client ke Foreign Agent 2 lalu berikan IP pada LAN

```
ip address 192.168.0.2
```

6. Tambahkan packet yang belum ada di CD/DVD menggunakan WinSCP

```
radvd
```

7. reboot PC Foreign Agent 2

Setelah proses *reboot* selesai lakukan konfigurasi

1. Login sebagai root

2. Lalu ketikkan modprobe ipv6

```
ForeignAgent2:~# modprobe ipv6
```

3. Setelah itu tambahkan IP address sesuai dengan topologi

```
ForeignAgen2t:~# nano /etc/network/interfaces
```

4. Tambahkan baris seperti berikut

```
auto lo
```

```
iface lo inet loopback
```

```
auto eth0
```

```
iface eth0 inet6 static
```

```
address 2001:0:0:b::3
```

```
netmask 64
```

```
gateway 2001:0:0:b::1
```

```
auto eth1
```

```
iface eth1 inet6 static
```

*address 2001:0:0::d::1*

*netmask 64*

( lalu keluar dengan menekan tombol ctrl + x jawab y lalu enter )

5. Setelah itu restart proses networknya dengan perintah

*ForeignAgent2:~# /etc/init.d/networking restart*

*echo 1 > /proc/sys/net/ipv6/conf/all/forwarding*

*route -A inet6 add 2001:0:0:a::/64 gw 2001:0:0:b::1 dev eth0*

*route -A inet6 add 2001:0:0:c::/64 gw 2001:0:0:b::2 dev eth0*

6. Kemudian lakukan instalasi paket *radvd* yang telah ditambahkan tadi

*ForeignAgent2:~# dpkg -i radvd.deb*

7. Lalu buat konfigurasi *radvd.conf* di */etc* dengan cara

*ForeignAgen2t:~# nano /etc/radvd.conf*

8. Lalu tambahkan baris berikut

*interface eth1*

*{*

*AdvSendAdvert on;*

*prefix 2001:0:0:d::/64*

*{*

*};*

*};*

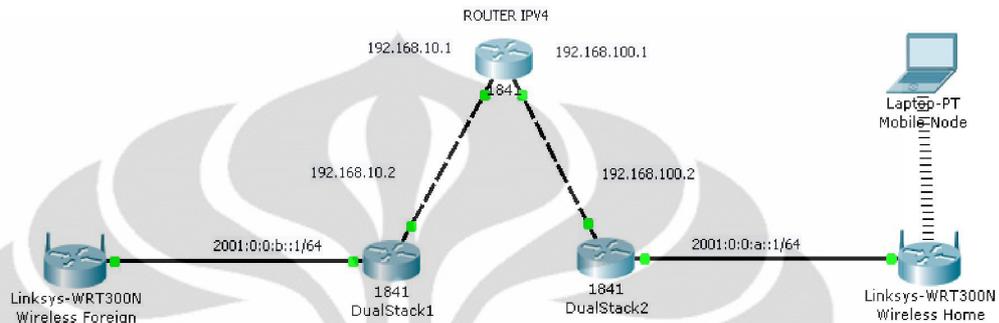
9. Lalu restart *radvd* dengan perintah

*ForeignAgent2:~# /etc/init.d/radvd restart*

Jika semua langkah di atas sudah selesai tinggal di coba melakukan test ping ke masing – masing PC

### 3.3 Konfigurasi Mobile Internet Protocol versi 6 (MIPv6) Tunneling 6to4

#### 3.3.1 Topologi



Gambar 3.2 Topologi MIPv6 *tunneling 6to4*

Pada gambar 3.2 merupakan gambaran topologi jaringan mobileIPv6 tunnelling 6to4. Pada Eth0 router IPv4 dihubungkan dengan Eth0 dari *dualstack 1* menggunakan kabel UTP cross, sementara Eth1 dihubungkan dengan Eth0 menggunakan *dualstack 2*. Pada Et1 dualstack 1 dihubungkan dengan *access point home (wireless home)*. Dan pada Eth1 dualstack 2 dihubungkan dengan *foreign (wireless foreign)*.

#### 3.3.2 Konfigurasi Router IPv4

1. Login sebagai root
2. Setelah itu tambahkan IP address sesuai dengan topologi

```
nano /etc/network/interfaces
```

3. Tambahkan baris seperti berikut

```
auto lo
```

```
iface lo inet loopback
```

```
auto eth0
```

```
iface eth0 inet static  
address 192.168.10.1  
netmask 255.255.255.0  
network 192.168.10.0  
broadcast 192.168.10.255
```

```
auto eth1
```

```
iface eth1 inet static  
address 192.168.100.1  
netmask 255.255.255.0  
network 192.168.100.0  
broadcast 192.168.100.255
```

( lalu keluar dengan menekan tombol ctrl + x jawab y lalu enter )

4. Setelah itu restart proses networknya dengan perintah

```
/etc/init.d/networking restart  
echo 1 > /proc/sys/net/ipv4/ip_forward
```

### 3.3.3 Konfigurasi di Home Agent

1. Login sebagai root
2. Setelah itu tambahkan IP address sesuai dengan topologi

```
nano /etc/network/interfaces
```

3. Tambahkan baris seperti berikut

```
auto lo
```

```
iface lo inet loopback
```

```
auto eth0
```

```
iface eth0 inet static
```

```
address 192.168.100.2  
netmask 255.255.255.0  
network 192.168.100.0  
broadcast 192.168.100.255  
gateway 192.168.100.1
```

```
auto eth1
```

```
iface eth1 inet6 static
```

```
address 2001:0:0:a::1
```

```
netmask 64
```

( lalu keluar dengan menekan tombol ctrl + x jawab y lalu enter )

4. Setelah itu tambahkan proses networknya dengan perintah

```
ifconfig sit0 add ::192.168.100.2/96 up  
/etc/init.d/networking restart  
echo 1 > /proc/sys/net/ipv4/ip_forward  
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding  
route -A inet6 add 2001:0:0:b::/64 gw ::192.168.10.2 dev sit0
```

### 3.3.4 Konfigurasi di Foreign Agent 1

1. Login sebagai root
2. Mount Paket dari CD/DVD

```
apt-cdrom add
```

3. Install Paket ssh-server

```
apt-get install ssh-server
```

4. Berikan IPv4 terlebih dahulu untuk menambah packet yang tidak ada di CD/DVD

***ifconfig eth0 192.168.0.1 netmask 255.255.255.0 up***

5. Koneksikan PC client ke *Foreign Agent* lalu berikan IP pada LAN

***ip address 192.168.0.2***

6. Tambahkan packet yang belum ada di CD/DVD menggunakan WinSCP

***radvd***

7. reboot PC Foreign Agent

Setelah proses *reboot* selesai lakukan konfigurasi

5. Login sebagai root

6. Setelah itu tambahkan IP address sesuai dengan topologi

***nano /etc/network/interfaces***

7. Tambahkan baris seperti berikut

***auto lo***

***iface lo inet loopback***

***auto eth0***

***iface eth0 inet static***

***address 192.168.10.2***

***netmask 255.255.255.0***

***network 192.168.10.0***

***broadcast 192.168.10.255***

***gateway 192.168.10.1***

***auto eth1***

***iface eth1 inet6 static***

***address 2001:0:0:b::1***

***netmask 64***

( lalu keluar dengan menekan tombol ctrl + x jawab y lalu enter )

8. Setelah itu tambahkan proses networknya dengan perintah

```
Ifconfig sit0 add ::192.168.10.2/96 up
```

```
/etc/init.d/networking restart
```

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
```

```
route -A inet6 add 2001:0:0:a::/64 gw ::192.168.100.2 dev sit0
```

9. Kemudian lakukan instalasi paket *radvd* yang telah ditambahkan sebelumnya

```
dpkg -i radvd.deb
```

10. Lalu buat konfigurasi *radvd.conf* di */etc* dengan cara

```
nano /etc/radvd.conf
```

11. Lalu tambahkan baris berikut

```
interface eth1
```

```
{
```

```
AdvSendAdvert on;
```

```
prefix 2001:0:0:b::/64
```

```
{
```

```
};
```

```
};
```

12. Lalu kita restart radvd dengan perintah

```
/etc/init.d/radvd restart
```

Jika semua langkah di atas sudah selesai tinggal di coba melakukan test ping

### 3.4 Metode Pengambilan Data

Pengambilan data dilakukan dengan cara :

1. Pada konfigurasi Mobile IPV6
  - a. Melakukan upload file dalam bentuk Win Rar dari Mobile (MN) node ke home agent (HA) pada saat MN terhubung dengan *access point* HA selanjutnya dipindah ke Acces point Foreign agent (FA).
  - b. Melakukan download file dalam bentuk Win Rar dari MN ke HA pada saat MN terhubung dengan *access point* HA kemudian kita pindah ke *access point* FA.
2. Pada konfigurasi tunneling 6to4 mobile IPv6
  - a. Melakukan upload file dalam bentuk Win Rar dari MN ke HA (*dualstack 1*) pada saat MN terhubung dengan *access point* HA selanjutnya dipindah ke *access point* FA.
  - b. Melakukan download file dalam bentuk Win Rar dari MN ke HA pada saat MN terhubung dengan *access point* HA selanjutnya dipindah ke *access point* FA (dual Stack 2).

Untuk proses pengambilan data akan dilakukan penangkapan paket-paket yang lewat dari sisi *mobile node* dengan menggunakan aplikasi *Wireshark*.

Terdapat tiga parameter yang akan diambil dalam pengambilan data yaitu *transfer time*, *trouput*, dan *delay*. Parameter tersebut dianggap dapat mewakili unjuk kerja dari SFTP dalam melakukan proses upload dan download data. Jika dilihat dari ketiga parameter tersebut memiliki ketrkaitan satu dengan lainnya.

File yang diupload dan download pada SFTP dibedakan menjadi dalam bermacam ukuran, yaitu 190Mbyte, 260Mbyte, dan 440Mbyte. Perbedaan ukuran file ini bertujuan untuk mengamati hubungan antara ukuran file ini bertujuan untuk mengamati hubungan antara ukuran file dengan parameter-parameter yang diuji.

Pada sekripsi ini dilakukan pengambilan data melalui IPv6 murni dan IPv6 dengan tunneling 6to4.

Penangkapan-penangkapan data melalui aplikasi *Wiresark* pada sisi *mobile node* . Pada proses penangkapan (*capture*) data tersebut akan

menampilkan banyak paket yang diterima, maka dari itu dilakukan proses filtering, sehingga yang akan terlihat hanya paket-paket secure file transfer protocol (SFTP) saja. Kemudian dapat dilihat pada *summary* dan mendapat parameter-parameter yang diinginkan, yaitu *transfer time*, *trouput* dan *delay*.



## **BAB IV**

### **ANALISA DATA**

#### **4.1 Analisa Konfigurasi Jaringan**

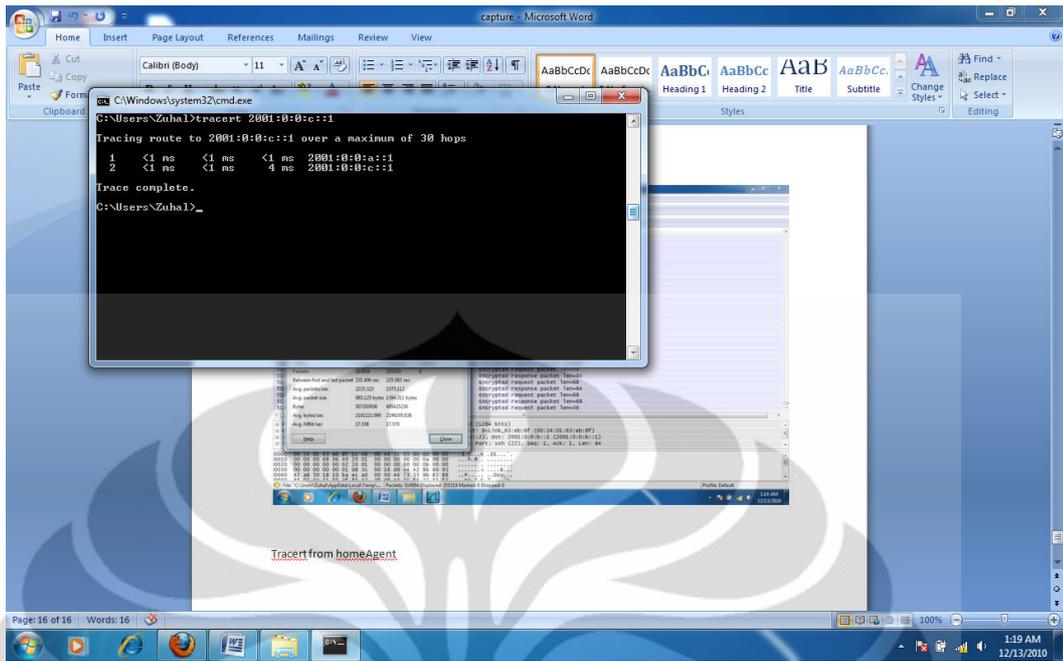
Pengambilan data dilakukan dengan menggunakan jaringan sederhana pada setiap konfigurasi jaringan yang diujikan. Secara keseluruhan untuk topologi yang digunakan sebuah laptop sebagai host/node mobile dan enam buah PC router sebagai *home agent-foreign agent* serta *router tunneling 6 to 4*. Masing-masing laptop di tambahkan aplikasi WinSCP dan Wireshark.

Pada penelitian ini dilakukan 2 percobaan yaitu menggunakan jaringan mobile IPv6 dan Mobile Ipv6 dengan tunneling dual stack .

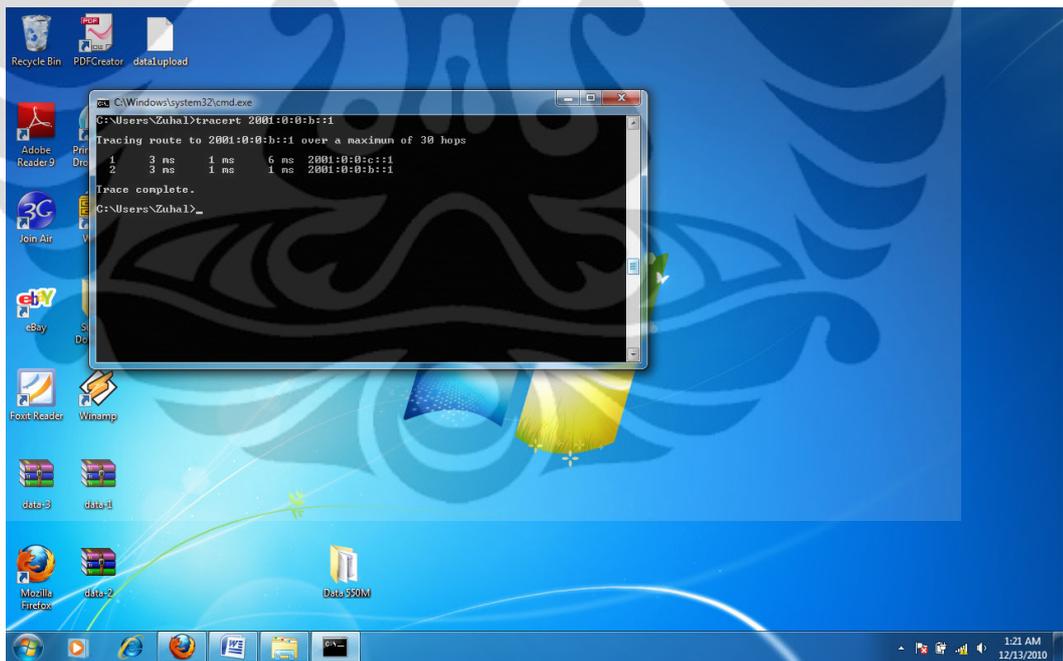
##### **4.1.1 Konfigurasi IPv6**

Pada konfigurasi jaringan mobile IPv6 seluruh perangkat baik router PC maupun mobile node diberikan alamat IPv6. Dalam konfigurasi routing yang digunakan adalah static routing, dikarenakan semua konfigurasi menggunakan alamat IPv6 maka proses transmisi data yang terjadi pada saat melalui router hanya routing dan forwarding paket seperti jaringan pada umumnya . Untuk konfigurasi lengkapnya dapat dilihat pada bab 3.

Hasil traceroute konfigurasi MIPv6 ditunjukkan pada Gambar 4.1 dari home agent ke foreign agent, dan untuk sebaliknya dari foreign agent ke home agent ditunjukkan pada Gambar 4.2 .Pada hasil tracerouter tersebut terlihat bahwa seluruh hop yang dilewati merupakan alamat IPv6.



Gambar 4.1 Tampilan *Tracerouter* dari *Home Agent*



Gambar 4.2 Tampilan *Tracerouter* dari *Foreign agent*

#### 4.1.2 Konfigurasi Mobile IPv6 dengan Tunneling 6to4

Pada konfigurasi jaringan tunnelling 6to4 juga memiliki topologi sama dengan mobile IPv6 sehingga tidak ada perbedaan jumlah hop. Pada konfigurasi ini menggunakan mekanisme tunnelling 6to4 sehingga router perlu ditambahkan interface tunnel yang di-setting menjadi mode 6to4.

Peran router pada konfigurasi ini selain dirouting, juga berfungsi untuk proses enkapsulasi dan dekapsulasi paket. Konsep tunneling adalah melewati IPv6 melalui jaringan IPv4. Akan tetapi alamat dari IPv4 diubah dahulu sesuai format alamat dari tunneling 6to4. Proses enkapsulasi ini berlangsung pada waktu melewati IPv6, melalui jaringan IPv4, dan setelah tiba di sisi seberang akan didekapsulasi agar tersisa alamat IPv6 saja sesuai dengan awal. Untuk konfigurasi lengkap jaringan tunneling 6to4 dapat dilihat pada bab 3.

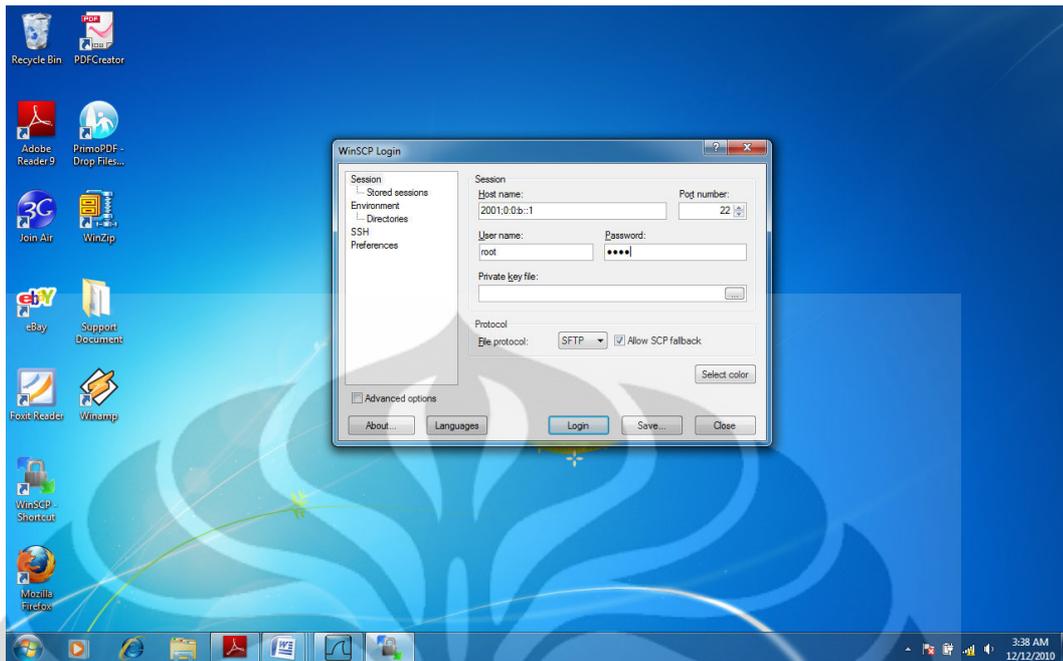
Hasil *traceroute* dilakukan dari sisi host ke host melalui tunnel. Dapat dilihat pada hasil *traceroute* bahwa pada hop kedua terdapat alamat IPv6. Alamat IPv6 pada hop kedua tersebut menunjukkan alamat *identifier* dari tunneling 6to4 sehingga dipastikan bahwa data yang dikirim dari host ke host melalui mekanisme tunneling 6to4 terlebih dahulu.

#### 4.2 Analisa Performa Jaringan Pada SFTP

Secure File Transfer Protocol (SFTP) merupakan aplikasi yang digunakan untuk memindahkan suatu file data dari host/mobile node. SFTP merupakan suatu jenis protokol yang bekerja dengan memanfaatkan protokol TCP/IP yang pada umumnya menggunakan port 22.

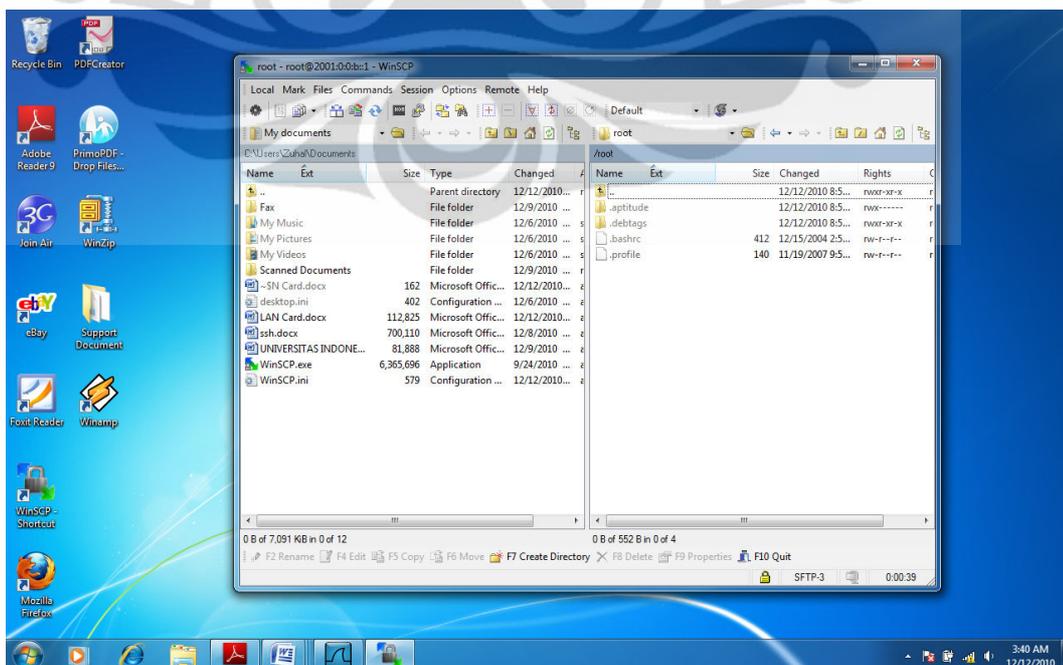
SFTP dibedakan jadi 2, yaitu proses upload dan download yang dilakukan mobile node.

Pada skripsi ini digunakan aplikasi WinSCP Pada semua PC dan mobile node. MIPv6 murni dan jaringan *tunneling* 6to4 akan diimplementasikan penggunaan aplikasi tersebut. Secara otomatis WinSCP dapat mendeteksi *address* yang ada pada masing-masing jaringan tersebut. Gambar 4.3 adalah aplikasi WinSCP.



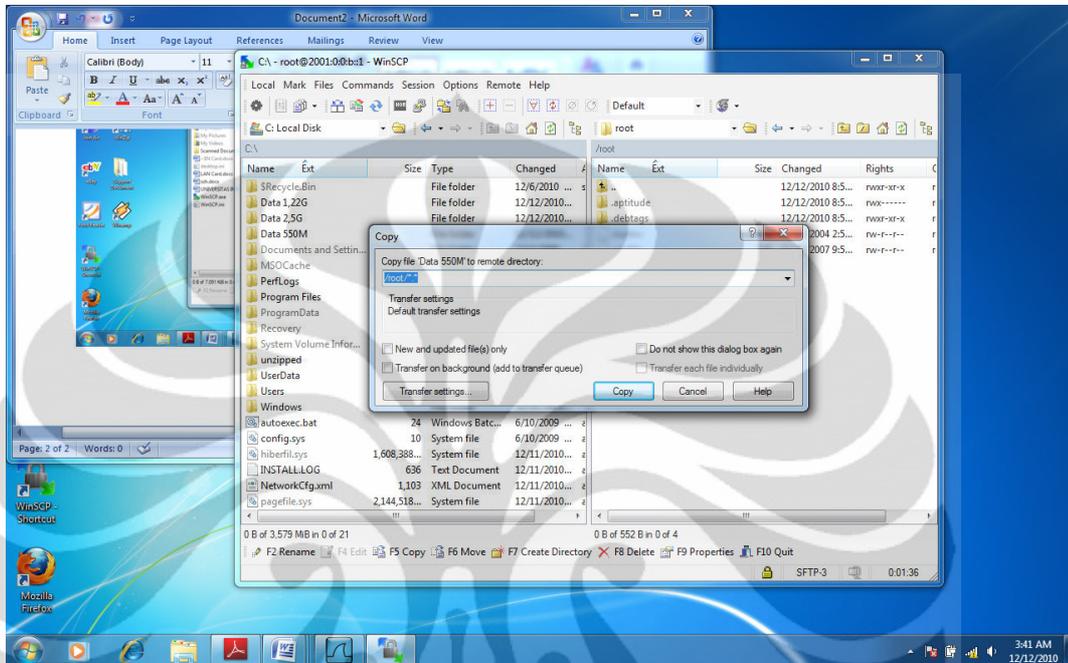
Gambar4.3 Tampilan aplikasi WinSCP

Setelah mobile node terhubung dengan accespoint home agent, maka selanjutnya akan diminta nomer IP Eth 0 , user name dan password home agent, setelah dimasukkan , pilih login. Apabila alamat yang dituju benar maka akan ditampilkan Gambar 4.4.



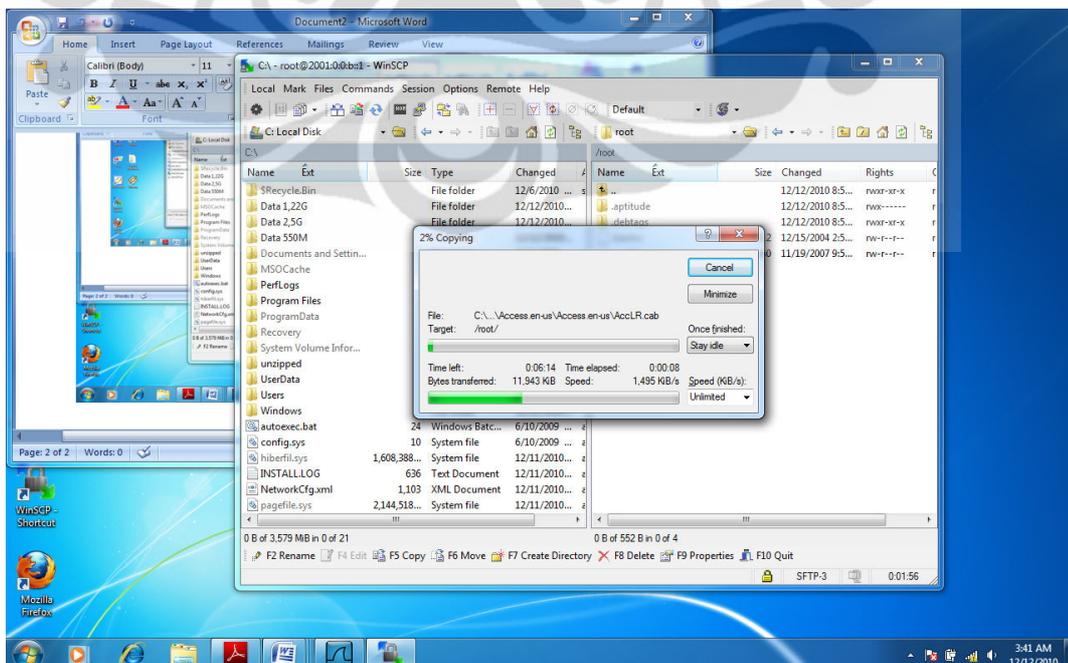
Gambar 4.4 Tampilan file yang tedapat pada MN dan HA

Selanjutnya proses upload maupun download dengan menekan file yang akan dipindahkan, baik dari mobile node maupun dari home agent dengan menekan menu copy seperti Gambar 4.5.



Gambar 4.5 tampilan eksekusi upload/download

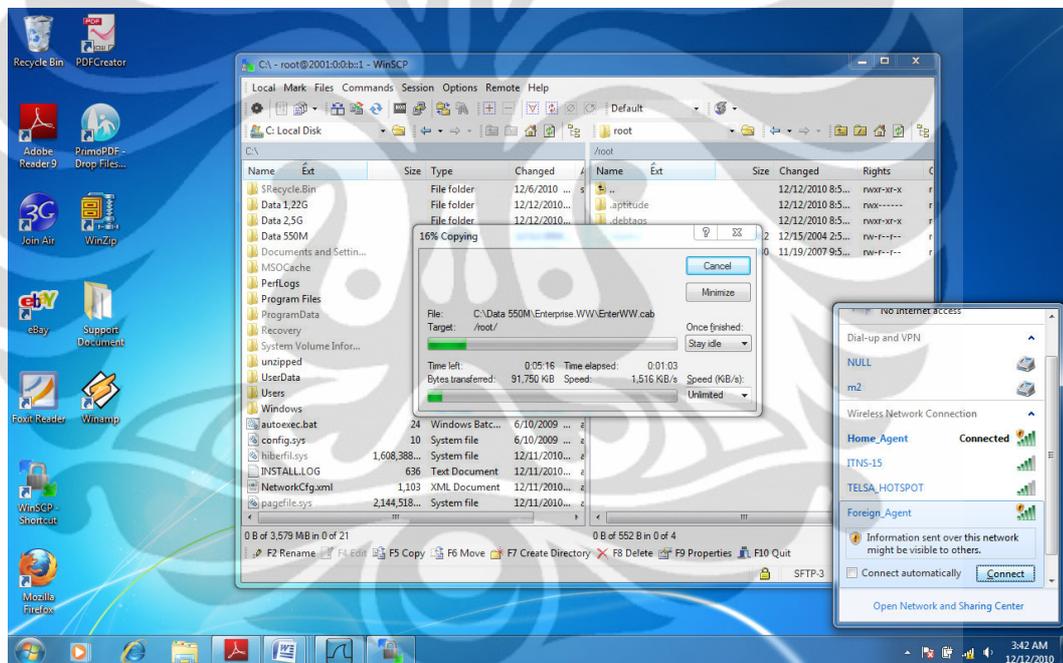
Proses upload dan download berlanjut seperti gambar 4.6



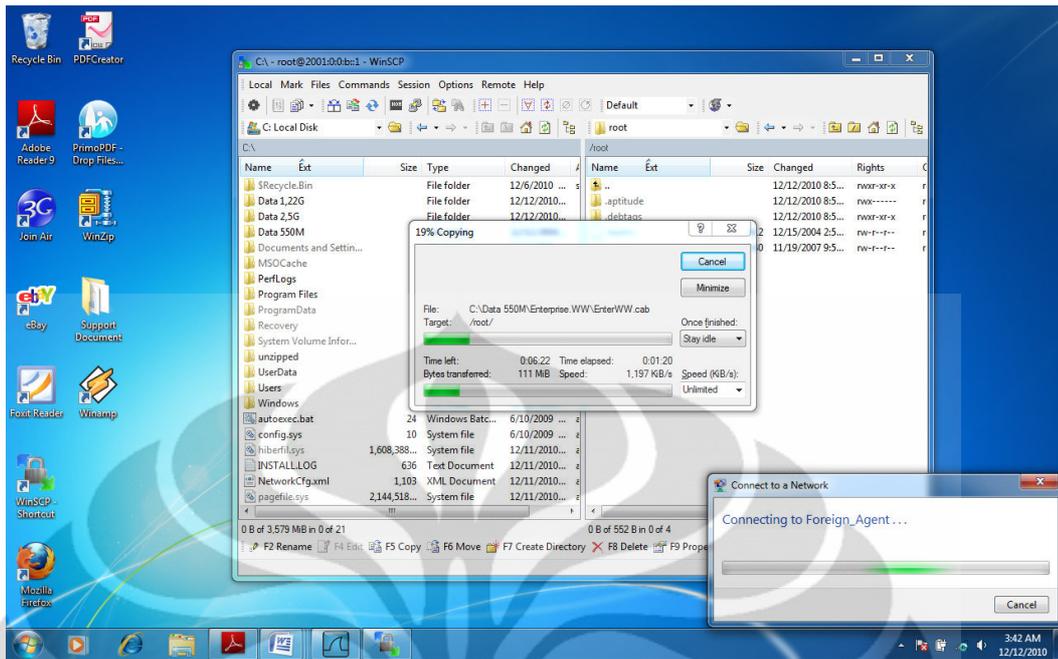
Gambar 4.6 Tampilan download/upload berlangsung

Proses selanjutnya mobil node pindah access point ke foreign agent dan diamati dan ditunggu prosesnya maka akan terjadi proses penghentian upload / download setelah tersambung lagi dan terkoneksi dengan foreign agent maka download /upload akan berlanjut.

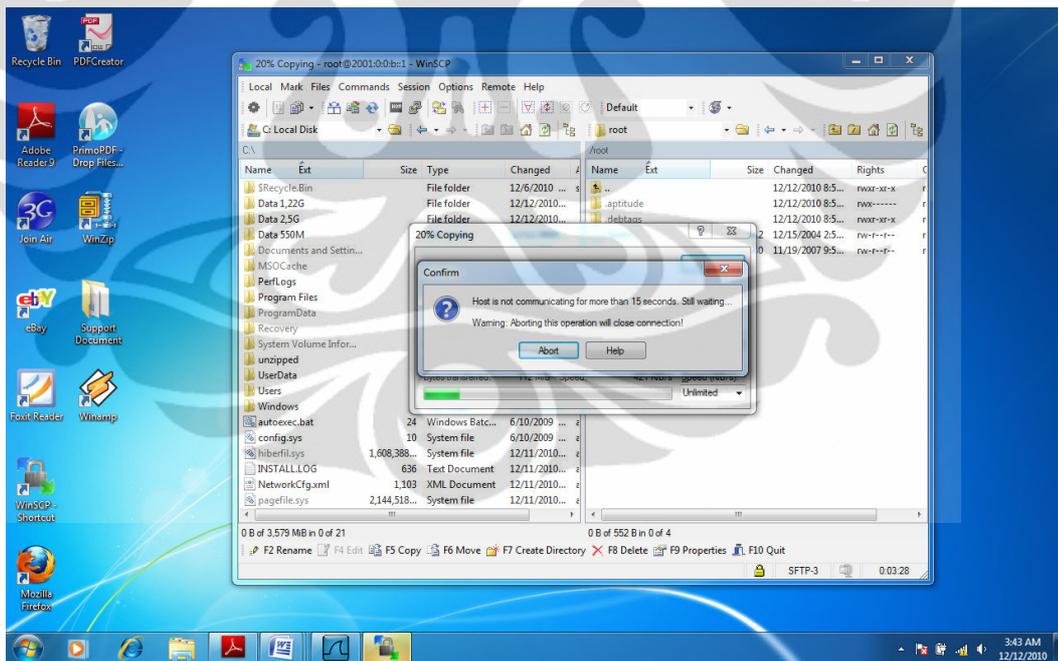
Proses dari berhentinya upload/download sampai berlanjutnya upload/download kembali, biasa disebut *Hand Over*. Waktu yang dibutuhkan untuk *hand over* berlangsung sekitar 10-15 detik, sebagai referensi dalam skripsi ini apabila menggunakan access point pada sisi *access point home agent (AH)* 300Mbps dan sisi access point foreign agent (AH) 54 Mbps maka akan menghasilkan *hand over* pada sisi AH lebih cepat sekitar 3detik dibanding pada sisi AF.



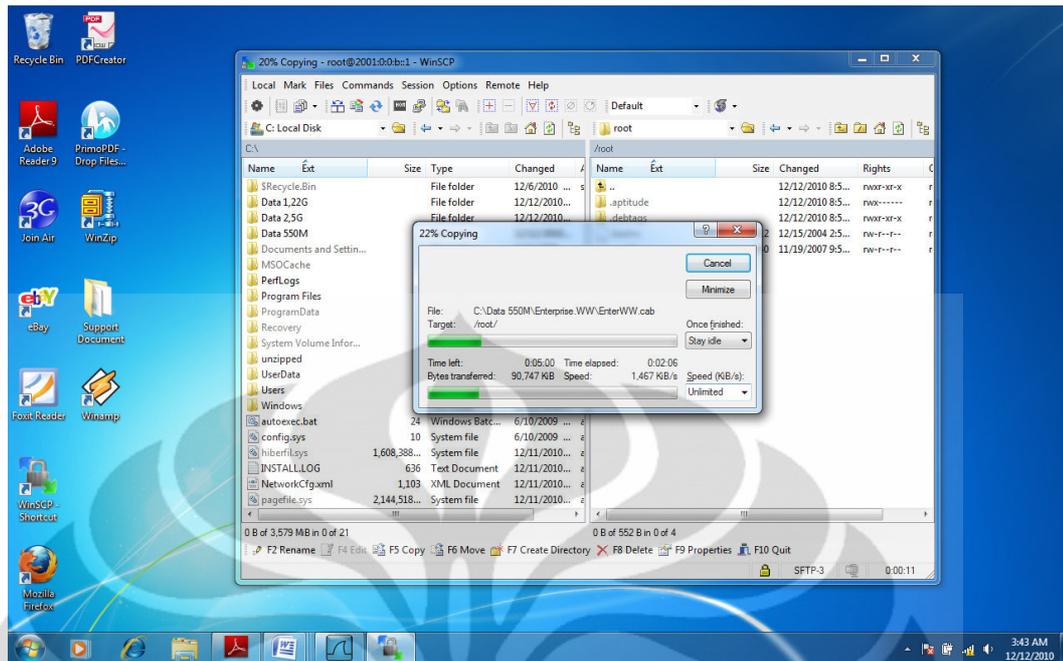
Gambar 4.7 Tampilan pindah accesspoint



Gambar 4.8 Tampilan proses koneksi access point baru



Gambar 4.9 tampilan proses hand over



Gambar 4.10 Tampilan proses upload/download setelah perpindahan koneksi dengan Access point baru

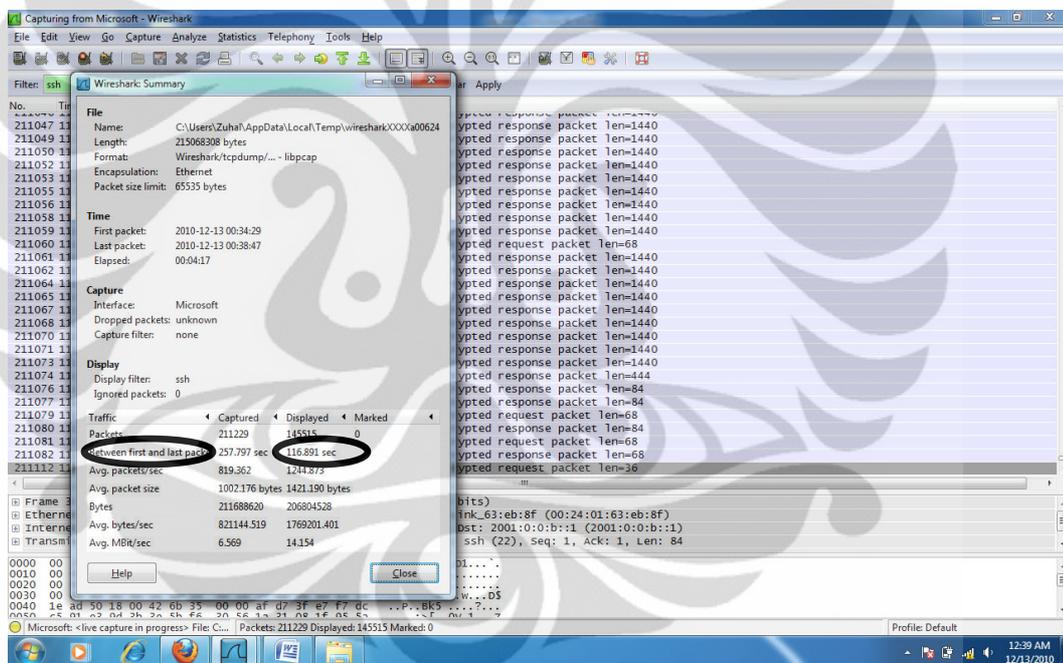
Proses download dan upload file dari dilakukan oleh mobile node ke Home Agent dengan ukuran yang berbeda-beda ukuran file : 190 MB, 260 MB, dan 440 MB. Semua file dibuat sama dalam bentuk ekstensi.rar untuk menghindari pengaruh perbedaan file pada performa SFTP.

Terdapat tiga parameter yang diambil dalam pengambilan data, yaitu *transfer time*, *throughput*, dan *delay*. Parameter tersebut dianggap mewakili unjuk kerja dari SFTP dalam melakukan proses down load dan download data. Jika dilihat dari ketiga parameter tersebut memiliki keterkaitan satu dengan yang lain. Untuk analisa data tentang ketiga parameter tersebut dijelaskan pada bagian berikut ini.

#### 4.2.1 Analisa Transfer Time

Transfer Time adalah jumlah waktu yang dibutuhkan untuk mengirim seluruh paket dari server ke client yang dinyatakan dalam second. Pengambilan parameter tranfertime dilakukandengan cara down load dan upload data dari server ke client. Kemudian disaat yang bersamaan pada sisi client atau mobile node melakukan capture data atau penangkapan paket-paket yang masuk melalui

interface ethernet dengan aplikasi wireshark. Berhubung paket-paket yang masuk pada ethernet tersebut bukan hanya paket SFTP, maka dari itu dilakukan filtering terlebih dahulu pada hasil summary capture wireshark sehingga hanya muncul bagian-bagian yang diinginkan saja. Pengambilan nilai transfer time pada summary Wireshark pada Gambar 4.11. Pengambilan nilai *transfer time* pada *summary* Wireshark dapat dilihat pada gambar di atas data yang digunakan hanya data paket yang displayed bukan pada captured. Data transfer time diambil dari perbedaan rentang waktu antara paket pertama sampai dengan paket terakhir. Dari tabel 3 dapat dilihat pada waktu ukuran file semakin besar kapasitas data maka semakin besar pula nilai dari transfer time.



Gambar 4.11 Pengambilan nilai Transfer time

Dapat dilihat Gambar 4.6 data yang digunakan hanya data paket yang ada pada displayed bukan pada captured. Data transfer time diambil dari perbedaan rentang waktu antara paket pertama sampai dengan paket terakhir. Dari pengujian data yang dilakukan sebanyak 5 kali pada masing-masing file dan masing-masing konfigurasi didapat rata-rata dari nilai transfer time seperti pada Tabel 4.1.

Diagram perbandingan nilai throughput ada pada Gambar 4,14.

Tabel 4.1 Transfer time rata-rata MIPv6

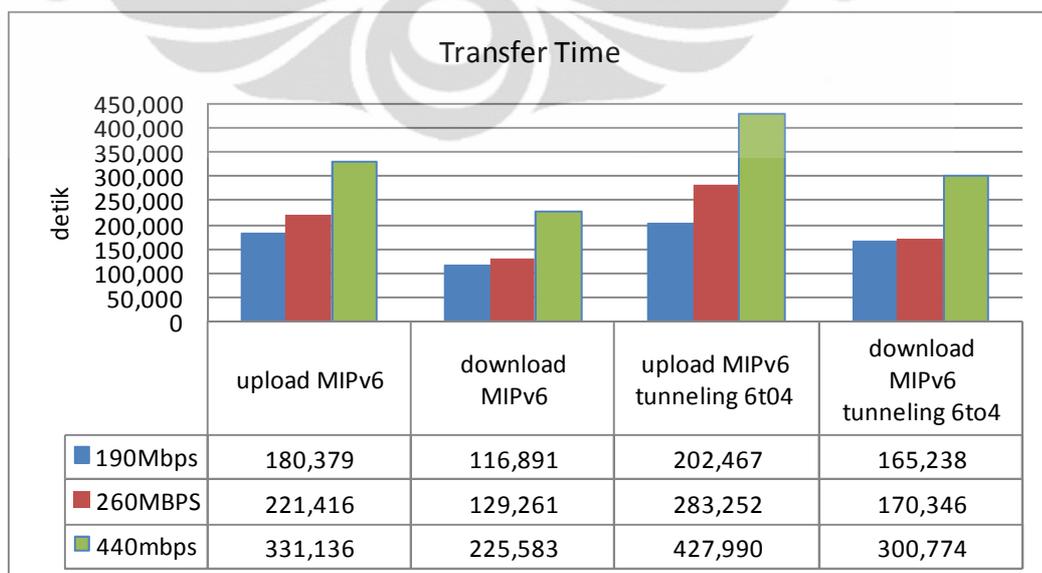
File (Mb)	Upload (second)	Download (second)
190	180,379	116,891
260	221,416	129,261
440	331,136	225,583

Tabel 4.2 Transfer time MIPv6 tunneling 6to4

File (Mb)	Upload (second)	Download (second)
190	202,467	165,238
260	283,252	170,346
440	427,990	300,774

Berdasarkan hasil rata rata pada Tabel 4.1 dan Tabel 4.2, didapatkan bahwa terdapat pengaruh kapasitas data terhadap nilai transfer time . Semakin besar kapasitas data maka semakin besar pula nilai dari transfer time.

Diagram perbandingan nilai transfer time dapat dilihat pada Gambar 4.12.



Gambar 4.12 Diagram Perbandingan Transfer

Untuk menghitung perbandingan transfer time saat upload dan download jaringan yang menggunakan konfigurasi MIPv6 dengan konfigurasi jaringan MIPv6 tunneling 6to4 dalam prosentase ditunjukkan rumus :

$$\% \text{PTtU} = \frac{\text{NTtU MIPv6 tunnel 6to4} - \text{NTtU MIPv6}}{\text{NTtU MIPv6}} \times 100\% \quad 4.1$$

Keterangan :

%PTtU = Prosentasi Perbandingan Transfer time saat Upload

NTtU = Nilai Transfer time saat Upload

$$\% \text{PTtD} = \frac{\text{NTtD MIPv6 tunnel 6to4} - \text{NTtD MIPv6}}{\text{NTtD MIPv6}} \times 100\% \quad 4.2$$

Keterangan :

%PTtD = Prosentasi Perbandingan Transfer time saat Download

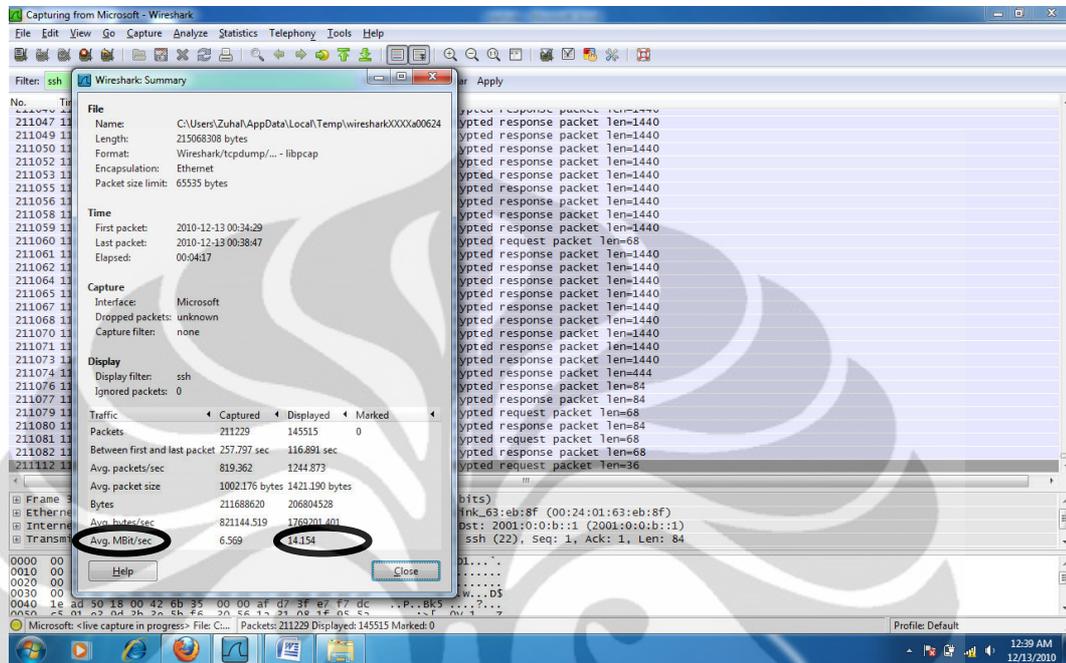
NTtD = Nilai Transfer time saat Download

Dari hasil perhitungan maka akan didapatkan nilai perbandingan transfer time saat upload pada konfigurasi MIPv6 lebih kecil antara 12% sampai 29% dibanding konfigurasi MIPv6 tunneling 6to4. Dan perbandingan transfer time saat download pada konfigurasi MIPv6 lebih kecil 31% sampai 41% dibanding konfigurasi MIPv6 tunneling 6to4. Hal ini dikarenakan adanya proses enkapsulasi dan dekapsulasi paket pada mekanisme tunneling. Selain routing, proses enkapsulasi dan dekapsulasi juga dapat menambah waktu transfer time suatu file.

#### 4.2.2 Analisa Throughput

Throughput merupakan kecepatan transfer data rata-rata dari suksesnya paket yang dikirim per detik, pada umumnya menggunakan satuan bit per second (bps). Pengambilan parameter transfer time dilakukan dengan cara download atau upload file dari mobile ke home dengan berpindah-pindah access point. Kemudian saat bersamaan pada sisi mobile node melakukan capture data atau penangkapan paket - paket yang masuk melalui *interface ethernet* dengan aplikasi Wireshark. Berhubung paket - paket yang masuk pada *ethernet* tersebut

bukan hanya paket SFTP, maka dari itu dilakukan *filtering* terlebih dahulu pada hasil *capture* Wireshark sehingga hanya muncul paket - paket SFTP saja.



Gambar 4.13 Pengambilan Data Throughput

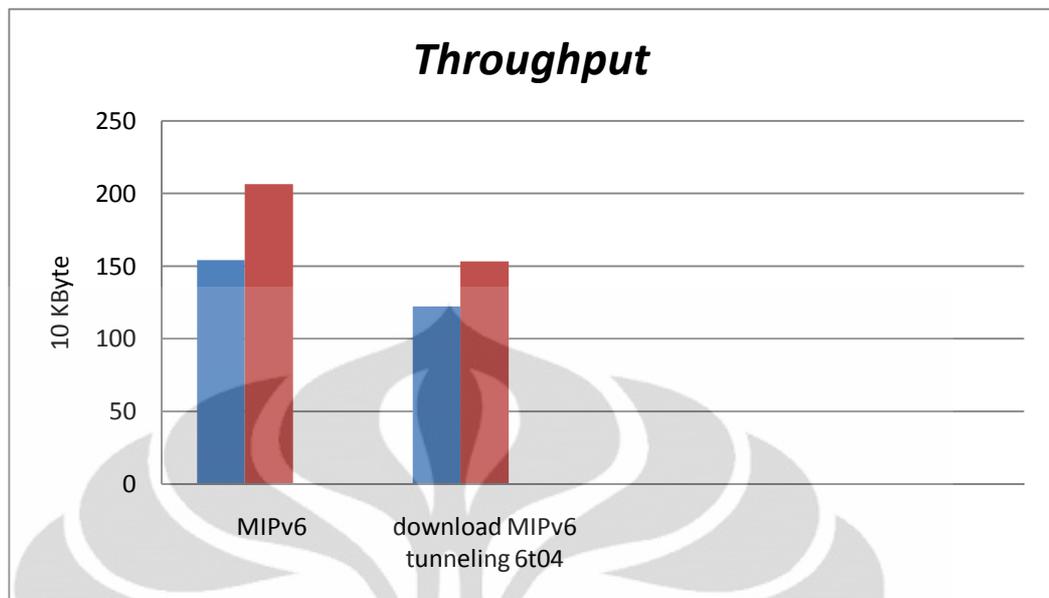
Dapat dilihat pada Gambar 4.13 data yang digunakan hanya data paket yang ada pada displayed bukan pada *captured*. Hal ini dimaksudkan agar yang ditampilkan hanya bagian – bagian yang diinginkan saja.

Tabel 4.3 Throughput MIPv6

File (Mb)	Upload (Mbps)	Download (Mbps)
190	1,31	1,77
260	1,49	2,23
440	1,82	2,20
Rata-rata	1,54	2,06

Tabel 4.4 Throughput MIPv6 tunneling 6to4

File (Mb)	Upload (Mbps)	Download (Mbps)
190	1,17	1,25
260	1,17	1,69
440	1,33	1,65
Rata-rata	1,22	1,53



Gambar 4.14 Diagram Perbandingan Throughput

Untuk menghitung perbandingan throughput saat upload dan download jaringan yang menggunakan konfigurasi MIPv6 dengan konfigurasi jaringan MIPv6 tunneling 6to4 dalam prosentase ditunjukkan rumus :

$$\% PThU = \frac{NThU \text{ MIPv6} - NThU \text{ MIPv6 tunnel 6to4}}{NThU \text{ MIPv6 tunnel 6to4}} \times 100\% \quad 4.3$$

Keterangan :

%PThU = Prosentasi Perbandingan Throughput saat Upload

NThU = Nilai Throughput saat Upload

$$\% PThD = \frac{NThD \text{ MIPv6} - NThD \text{ MIPv6 tunnel 6to4}}{NThD \text{ MIPv6 tunnel 6to4}} \times 100\% \quad 4.4$$

Keterangan :

%PThD = Prosentasi Perbandingan Throughput saat Download

NThD = Nilai Throughput saat Download

Dari hasil perhitungan maka akan didapatkan nilai perbandingan throughput saat upload pada konfigurasi MIPv6 lebih besar antara 12% sampai 36% dibanding konfigurasi MIPv6 tunneling 6to4. Dan perbandingan throughput saat download pada konfigurasi MIPv6 lebih besar 31% sampai 41% dibanding

konfigurasi MIPv6 tunneling 6to4. Hal ini dikarenakan adanya proses enkapsulasi dan dekapsulasi paket pada mekanisme tunneling. Selain routing, proses enkapsulasi dan dekapsulasi juga dapat mengurangi throughput suatu file.

### 4.2.3 Analisa Delay

*Delay* adalah waktu tunda dari waktu yang seharusnya dari suksesnya seluruh paket yang diterima. Parameter *delay* dihitung dengan cara membagi nilai *transfer time* dengan jumlah bit data. Ditunjukkan pada rumus dibawah ini :

$$delay \text{ (sec)} = \frac{transfer \text{ time (sec)}}{jumlah \text{ bit}} \quad 4.5$$

Hasil pengambilan data secara keseluruhan untuk parameter *delay* dapat dilihat pada bagian lampiran sedangkan untuk hasil rata – rata dari perhitungan *delay* dapat dilihat pada Tabel 4.5.

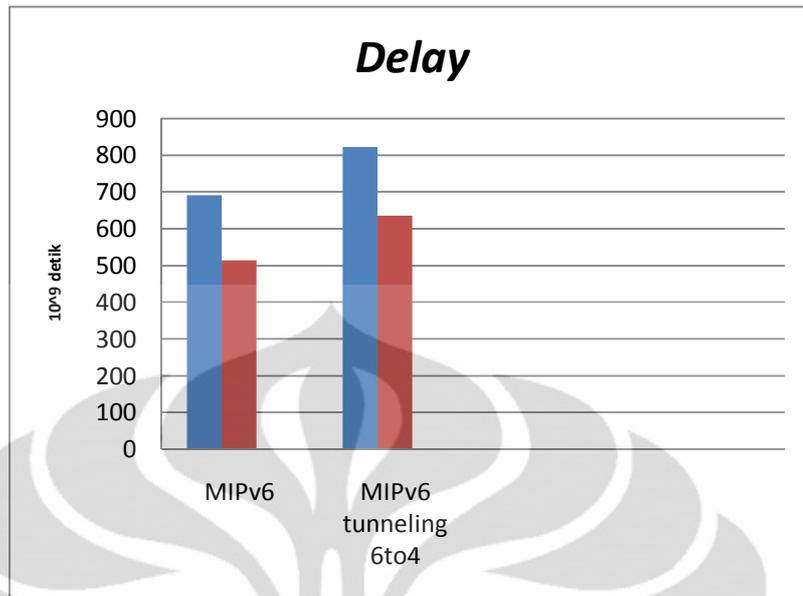
Tabel 4.5 Delay pada Jaringan MIPv6

File (Mb)	Upload (µs)	Download (µs)
190	0,761	0,565
260	0,761	0,458
440	0,548	0,516
Rata-rata	0,690	0,513

Tabel 4.6 Delay MIPv6 tunneling 6to4

File (Mb)	Upload (µs)	Download (µs)
190	0,854	0,799
260	0,857	0,590
440	0,754	0,606
Rata-rata	0,822	0,665

Untuk diagram perbandingan nilai delay dapat dilihat pada Gambar 4.15.



Gambar 4.15 Diagram Perbandingan Delay

Untuk menghitung perbandingan transfer time saat upload dan download jaringan yang menggunakan konfigurasi MIPv6 dengan konfigurasi jaringan MIPv6 tunneling 6to4 dalam prosentase ditunjukkan rumus :

$$\% \text{ PDU} = \frac{\text{NDU MIPv6 tunnel 6to4} - \text{NDU MIPv6}}{\text{NDU MIPv6}} \times 100\% \quad 4.6$$

Keterangan :

%PDU = Prosentasi Perbandingan Delay saat Upload

NDU = Nilai Delay saat Upload

$$\% \text{ PDD} = \frac{\text{NDD MIPv6 tunnel 6to4} - \text{NDD MIPv6}}{\text{NDD MIPv6}} \times 100\% \quad 4.7$$

Keterangan :

%PDD = Prosentasi Perbandingan Delay saat Download

NTtD = Nilai Dealay saat Download

Dari hasil perhitungan maka akan didapatkan nilai perbandingan delay saat upload pada konfigurasi MIPv6 lebih kecil antara 12% sampai 38% dibanding konfigurasi MIPv6 tunneling 6to4. Dan perbandingan transfer time

saat download pada konfigurasi MIPv6 lebih kecil 17% sampai 41% dibanding konfigurasi MIPv6 tunneling 6to4. Ini menunjukkan bahwa selain proses routing juga dapat menambah delay jaringan. Hal ini dikarenakan adanya proses enkapsulasi dan dekapsulasi paket-paket pada proses tunneling.



## BAB 5

### KESIMPULAN

- 1 Untuk jaringan yang menggunakan MIPv6 secara presentasi memiliki nilai transfer time rata-rata lebih kecil pada saat upload 12% sampai 29% dan saat download 31% sampai 41% dari konfigurasi jaringan MIPv6 tunneling 6to4. Hal ini dikarenakan adanya proses enkapsulasi dan dekapsulasi paket pada mekanisme tunneling. Selain routing, proses enkapsulasi dan dekapsulasi juga dapat menambah waktu transfer suatu file.
- 2 Konfigurasi MIPv6 memiliki nilai throughput lebih besar pada saat upload 12% sampai 36% dan pada saat download 31% sampai 41% dibanding konfigurasi MIPv6 tunneling 6to4, hal ini dikarenakan adanya proses enkapsulasi dan dekapsulasi paket pada mekanisme tunneling. Selain routing, proses enkapsulasi dan dekapsulasi paket juga dapat mengurangi kecepatan rata-rata kirim keseluruhan paket.
- 3 Untuk delay dari konfigurasi MIPv6 lebih kecil pada saat upload 12% sampai 38% dan pada saat download 17% sampai 41% dibanding MIPv6 tunneling 6to4. Ini menunjukkan bahwa selain proses routing juga dapat menambah delay jaringan. Hal ini dikarenakan adanya proses enkapsulasi dan dekapsulasi paket-paket yang masuk ada tunneling.

## DAFTAR REFERENSI

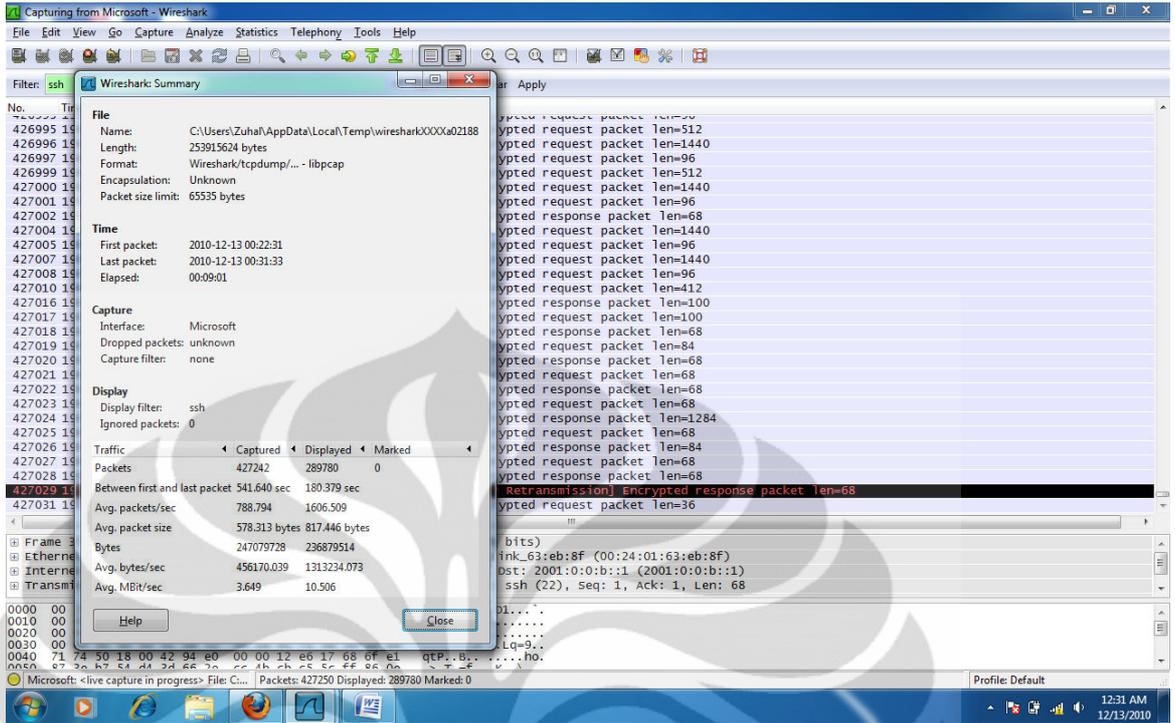
- [1] Mobile Networking Through.  
<http://Computer.Org/Internet/v2n1/perkins.htm>
- [2] Data and Computer Communications, Stallings, W. Practice Hall, 2000
- [3] Ghosh, Debalina. *Mobile IP*. <http://www.acm.org/crossroads/xrds7-2/mobileip.html>
- [4] *Mobile IP: Design Principles and Practice*, Addison-Wesley Longman, Reading, Mass., 1998.
- [5] Computer Networks, Andrew S. Tanenbaum, 2003
- [6] Linux Mobile IPv6 HOWTO 2004-04-20
- [7] Wikipedia, "Tunneling Protocol", Diakses Maret 2010 dari Wikipedia.  
[http://en.wikipedia.org/wiki/Tunneling\\_protocol](http://en.wikipedia.org/wiki/Tunneling_protocol)
- [8] Wikipedia, "6to4", Diakses April 2010 dari Wikipedia  
<http://en.wikipedia.org/wiki/6to4>
- [9] [en.wikipedia.org/wiki/SSH\\_File\\_Transfer\\_Protocol](http://en.wikipedia.org/wiki/SSH_File_Transfer_Protocol) diakses Desember 2010

# LAMPIRAN HASIL CAPTURE WIRESARK

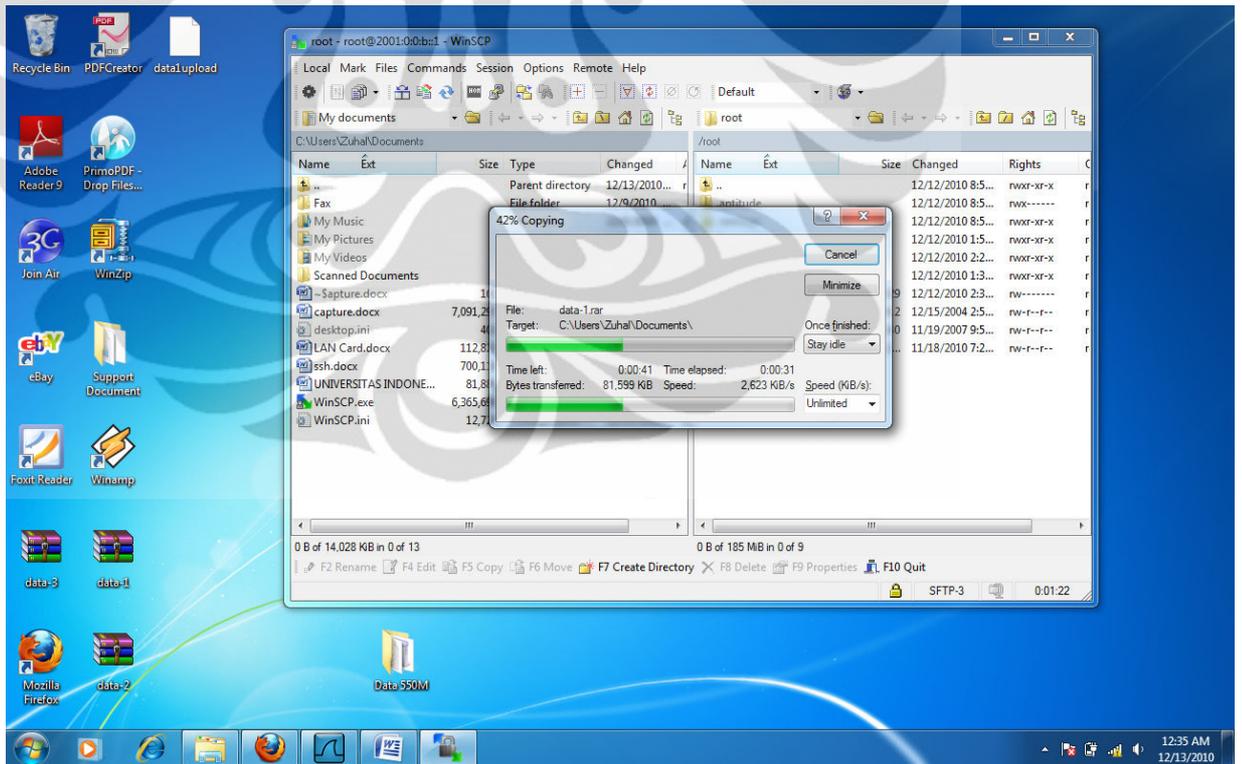
## UPLOAD FILE 190MB PADA JARINGAN MIPv6

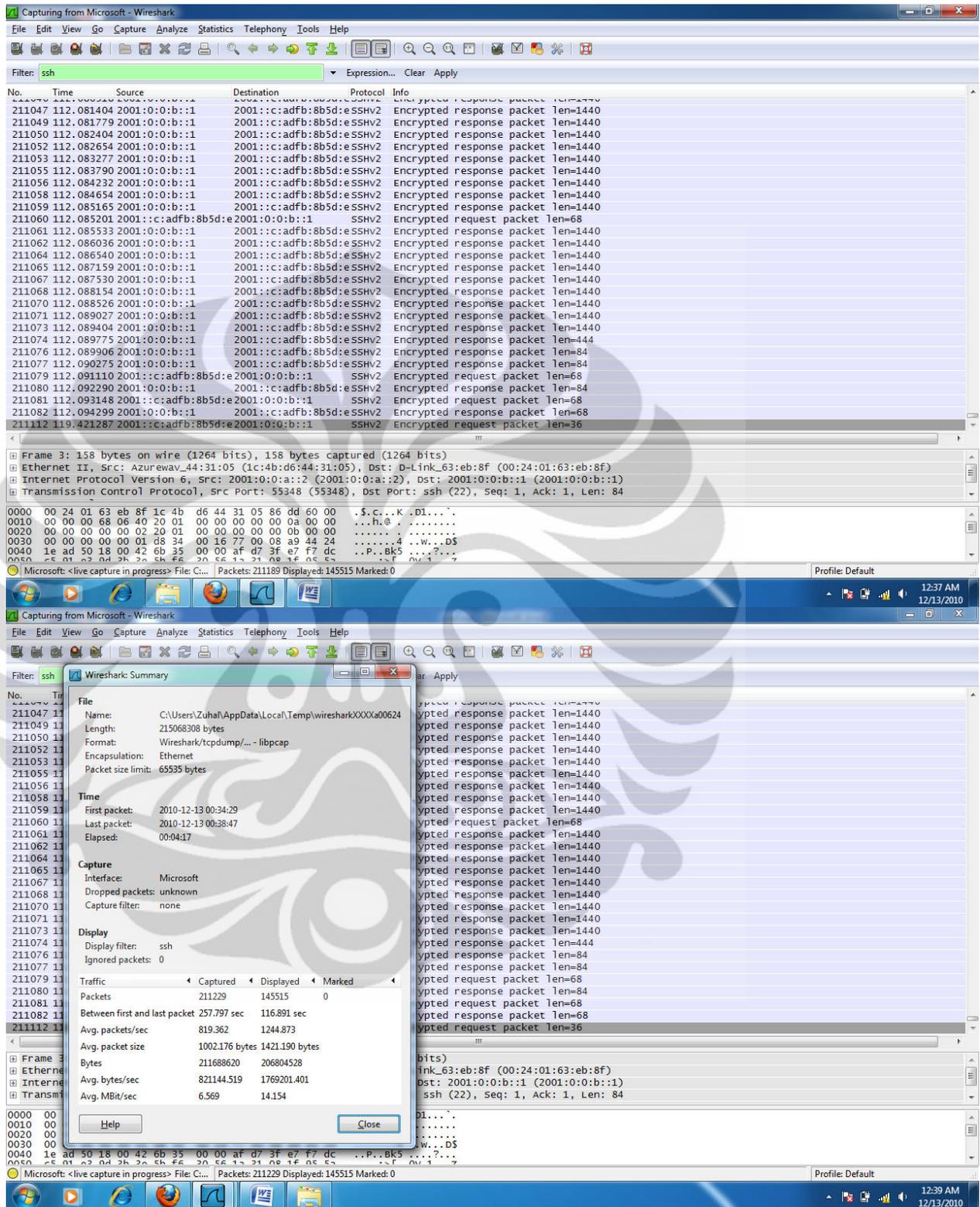
The screenshot displays a Windows desktop environment. In the foreground, a WinSCP window is open, showing a file upload progress dialog for 'data-1.rar' (194,913 KB) to the root directory. The dialog indicates a speed of 1,446 KB/s and a time elapsed of 0:00:05. Below the WinSCP window, the Wireshark network traffic capture is visible, filtered for 'ssh'. The packet list shows a retransmission of an encrypted response packet (len=68) at time 190.976936. The packet details pane shows the retransmission of an encrypted response packet (len=68) over Ethernet II, Internet Protocol Version 6, and Transmission Control Protocol. The status bar at the bottom of Wireshark shows 'Packets: 427198 Displayed: 289780 Marked: 0'.

(LANJUTAN)



## DOWNLOAD FILE 190M MIPv6



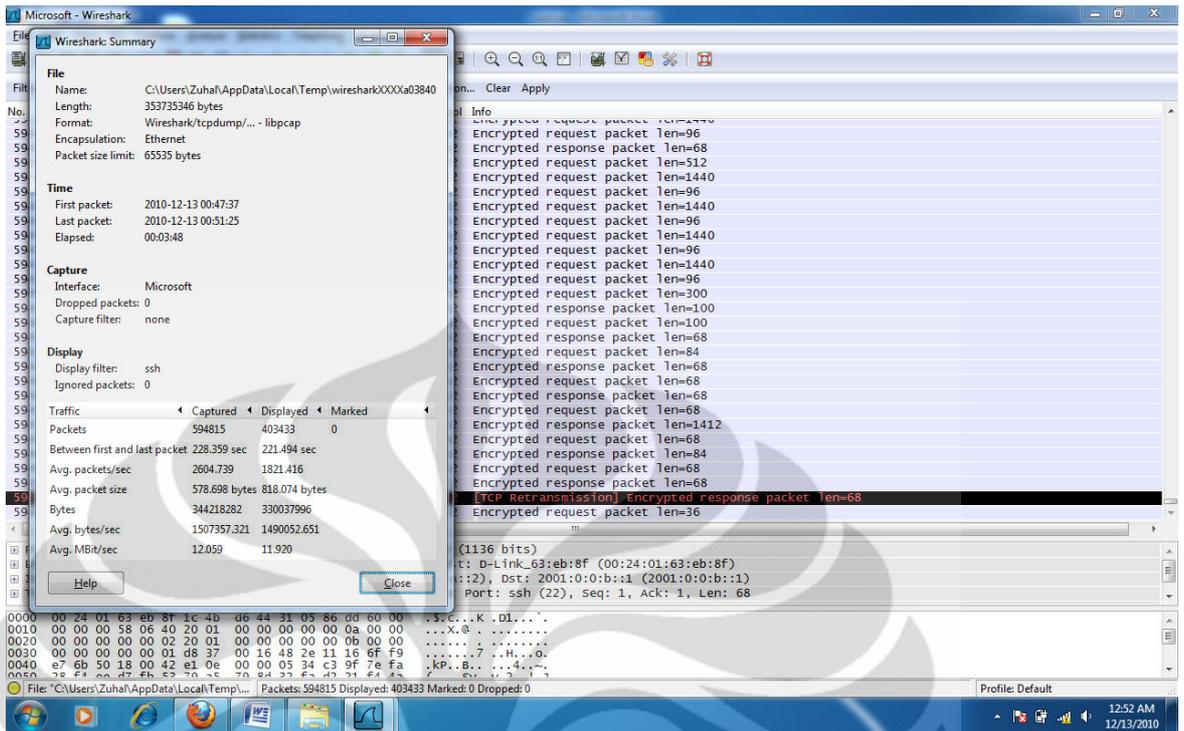


## UPLOAD FILE 260M MIPv6

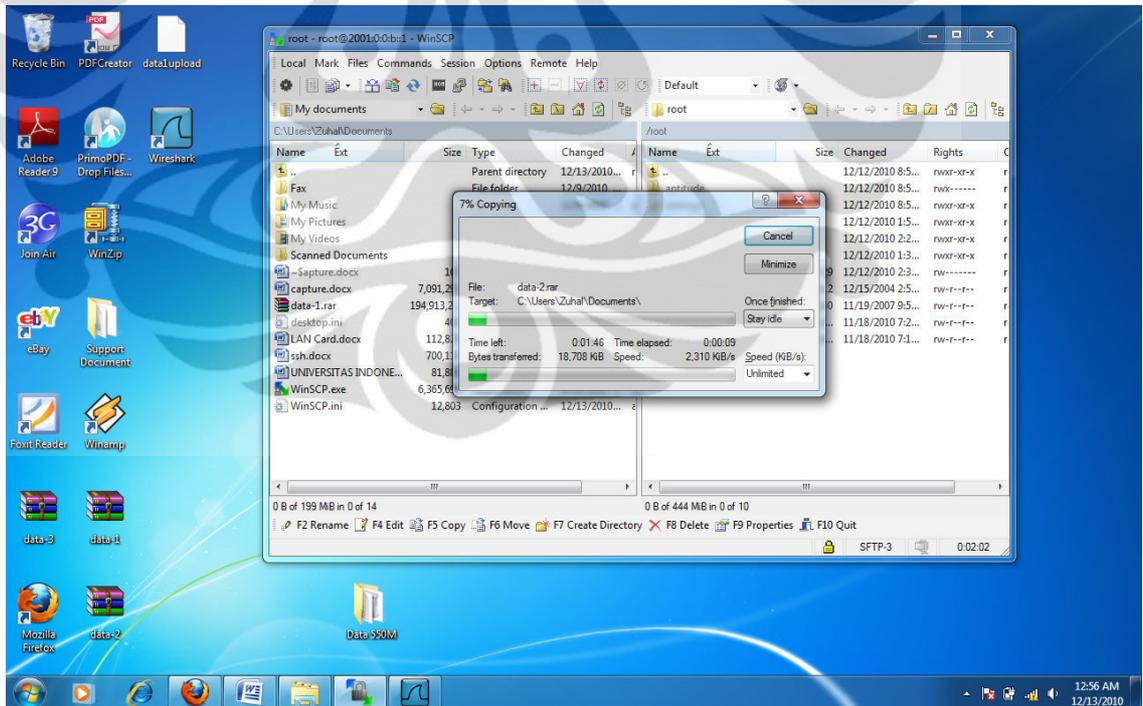
The screenshot shows a Windows XP desktop environment. In the foreground, a WinSCP window is open, displaying a file transfer progress dialog for '5% Copying'. The dialog shows the source file 'C:\Users\Zuhaf\Desktop\data-2.rar' being transferred to the target directory '/root/'. The progress bar indicates 5% completion, with 15,406 KB of bytes transferred at a speed of 1,549 KB/s. The estimated time left is 0:02:41, and the time elapsed is 0:00:10. The WinSCP window also shows a file list for the local desktop and the remote root directory.

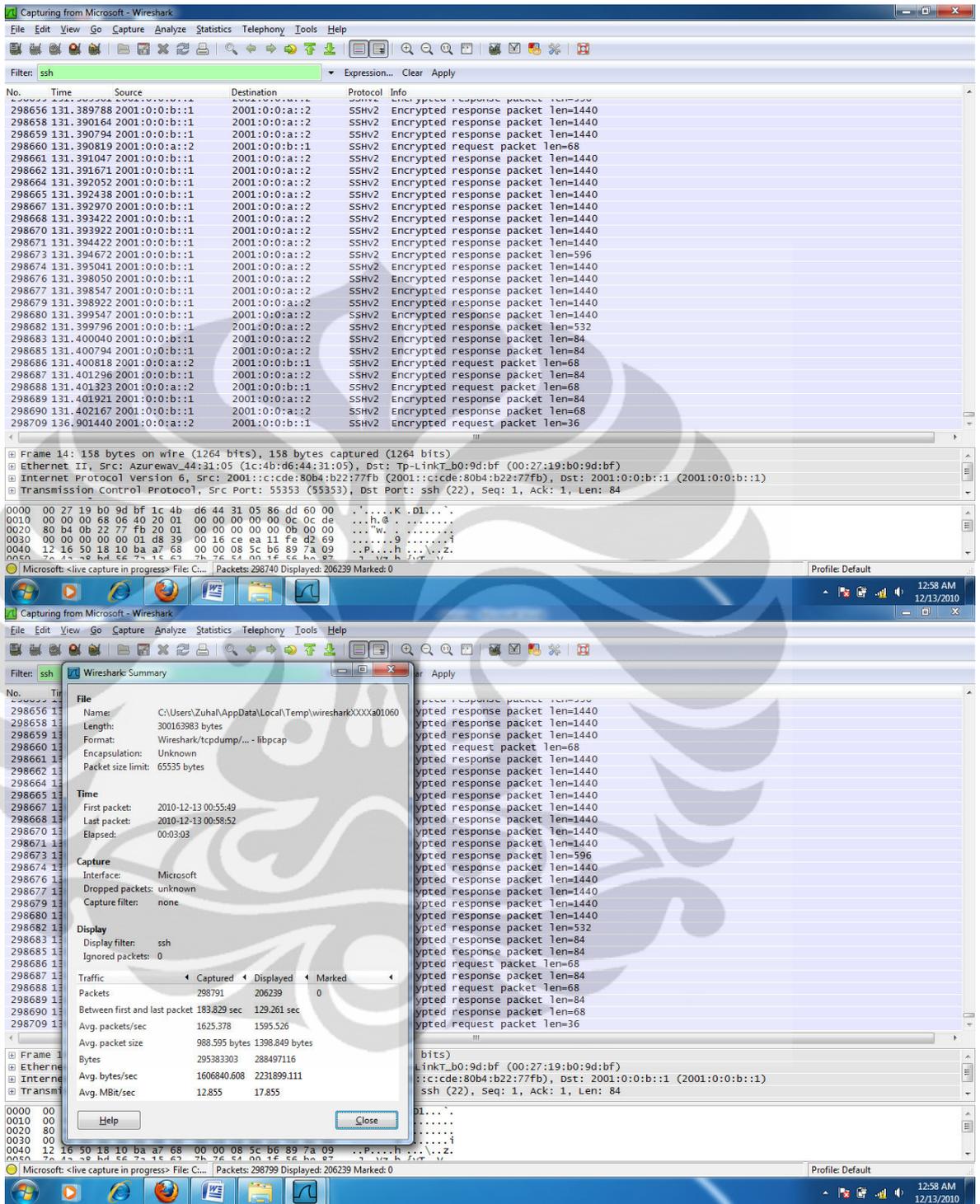
Below the WinSCP window, a Microsoft Wireshark window is open, displaying network traffic analysis. The filter is set to 'ssh'. The packet list shows several SSHV2 encrypted request and response packets. The packet details pane shows the structure of an SSHV2 packet, including the Ethernet II, Internet Protocol Version 6, and Transmission Control Protocol headers. The hex and ASCII views of the packet data are also visible.

No.	Time	Source	Destination	Protocol	Info
594769	220.837145	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=96
594770	220.838089	2001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted response packet len=68
594772	220.840863	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=512
594773	220.840901	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=1440
594774	220.840907	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=96
594776	220.841163	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=1440
594777	220.841170	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=96
594779	220.842378	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=1440
594780	220.842383	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=96
594782	220.842632	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=1440
594783	220.842639	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=96
594786	220.843370	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=300
594791	220.845956	2001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted response packet len=100
594792	220.846250	2001::c:cde:80b4:b22001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted request packet len=100
594793	220.847581	2001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted response packet len=68
594794	220.847704	2001::c:cde:80b4:b22001:0:0:b::1	2001:0:0:b::1	SSHV2	Encrypted request packet len=84
594795	220.849207	2001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted response packet len=68
594796	220.849847	2001::c:cde:80b4:b22001:0:0:b::1	2001:0:0:b::1	SSHV2	Encrypted request packet len=68
594797	220.851086	2001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted response packet len=68
594798	220.851342	2001::c:cde:80b4:b22001:0:0:b::1	2001:0:0:b::1	SSHV2	Encrypted request packet len=68
594799	220.854835	2001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted response packet len=1412
594800	220.855153	2001::c:cde:80b4:b22001:0:0:b::1	2001:0:0:b::1	SSHV2	Encrypted request packet len=68
594801	220.856462	2001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted response packet len=84
594802	220.856703	2001::c:cde:80b4:b22001:0:0:b::1	2001:0:0:b::1	SSHV2	Encrypted request packet len=68
594803	220.857966	2001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	Encrypted response packet len=68
594804	221.058384	2001:0:0:b::1	2001::c:cde:80b4:b22001:0:0:b::1	SSHV2	[TCP Retransmission] Encrypted response packet len=68
594812	228.355224	2001::c:cde:80b4:b22001:0:0:b::1	2001:0:0:b::1	SSHV2	Encrypted request packet len=36

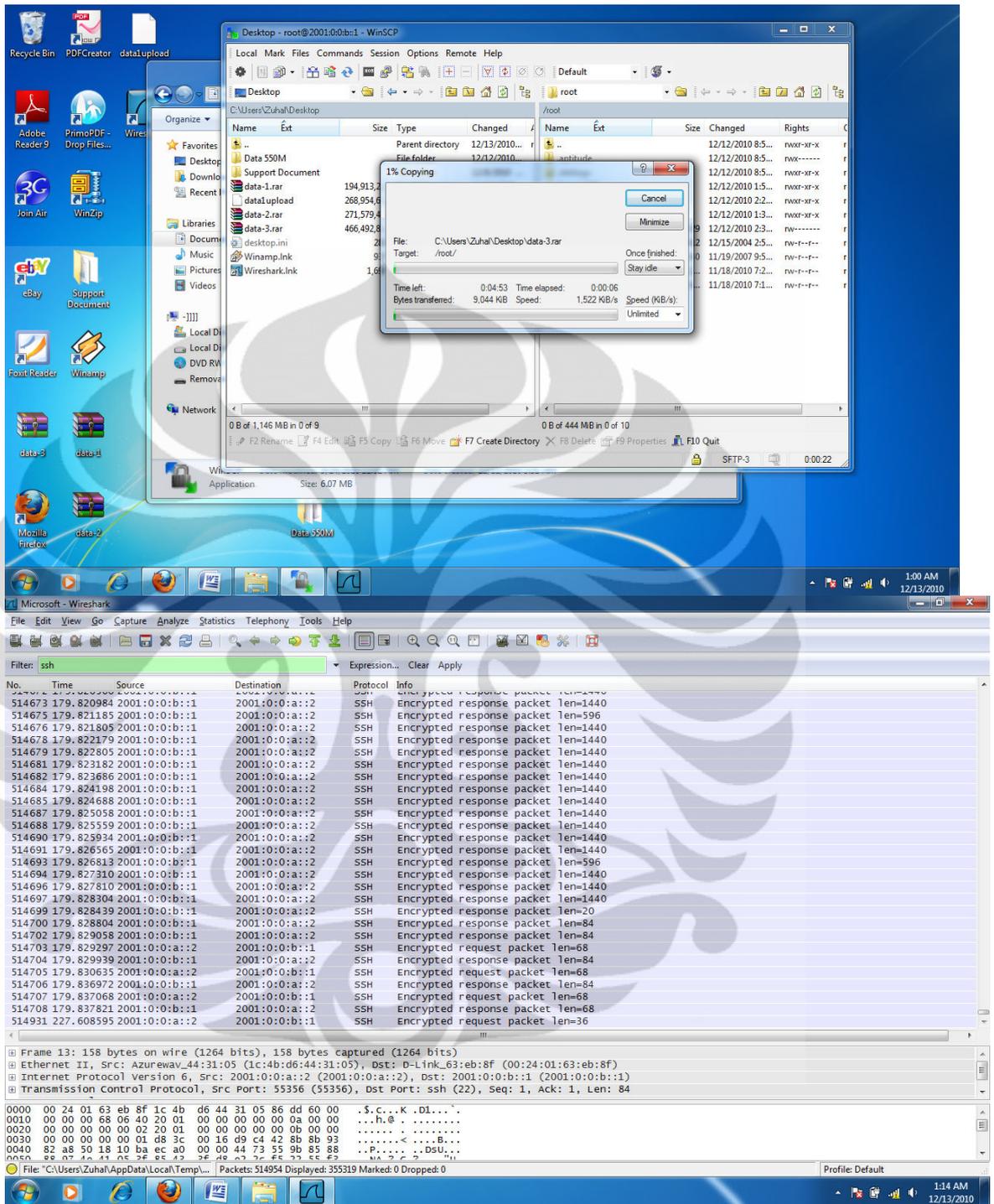


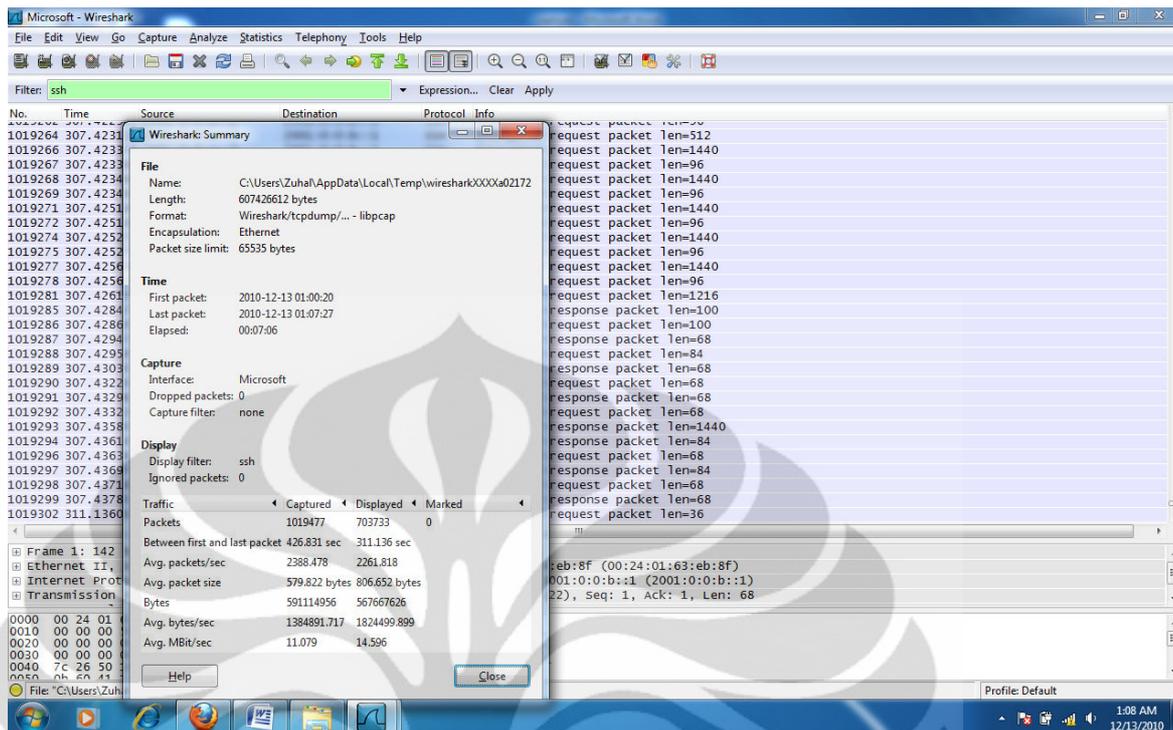
## DOWNLOAD FILE 260 MIPV6



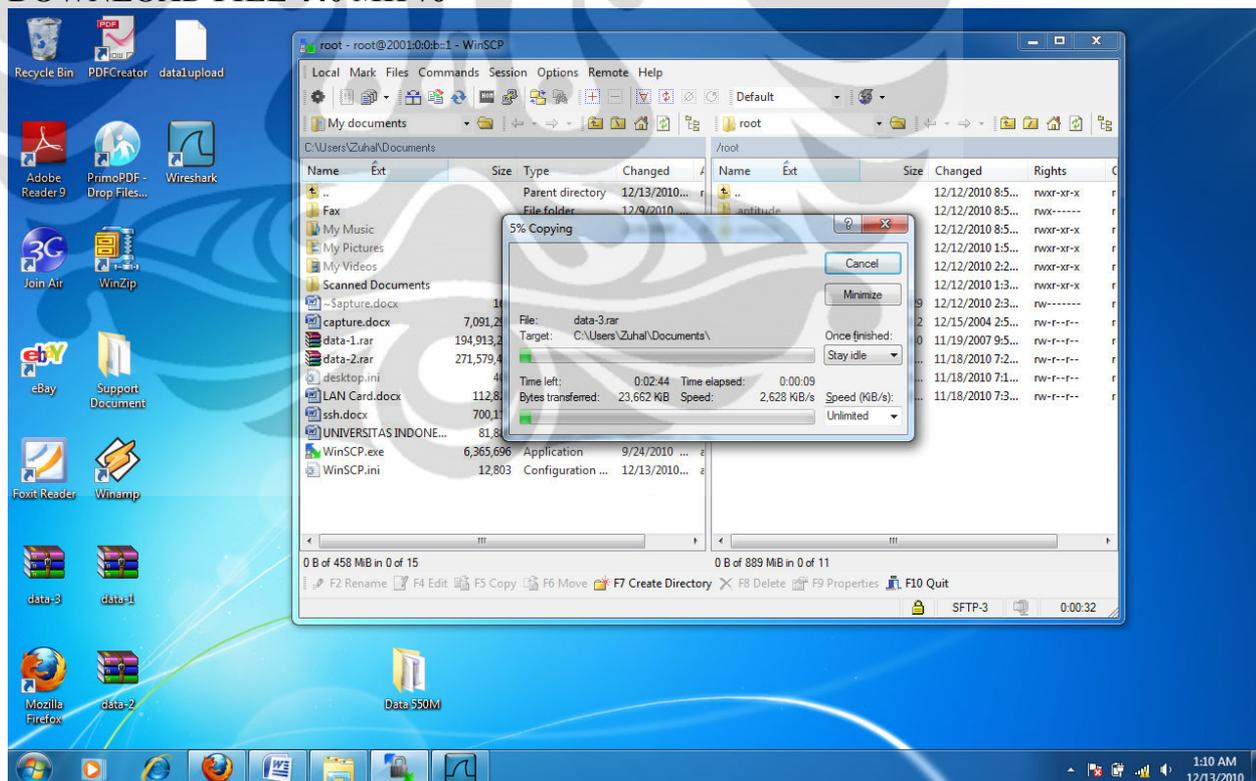


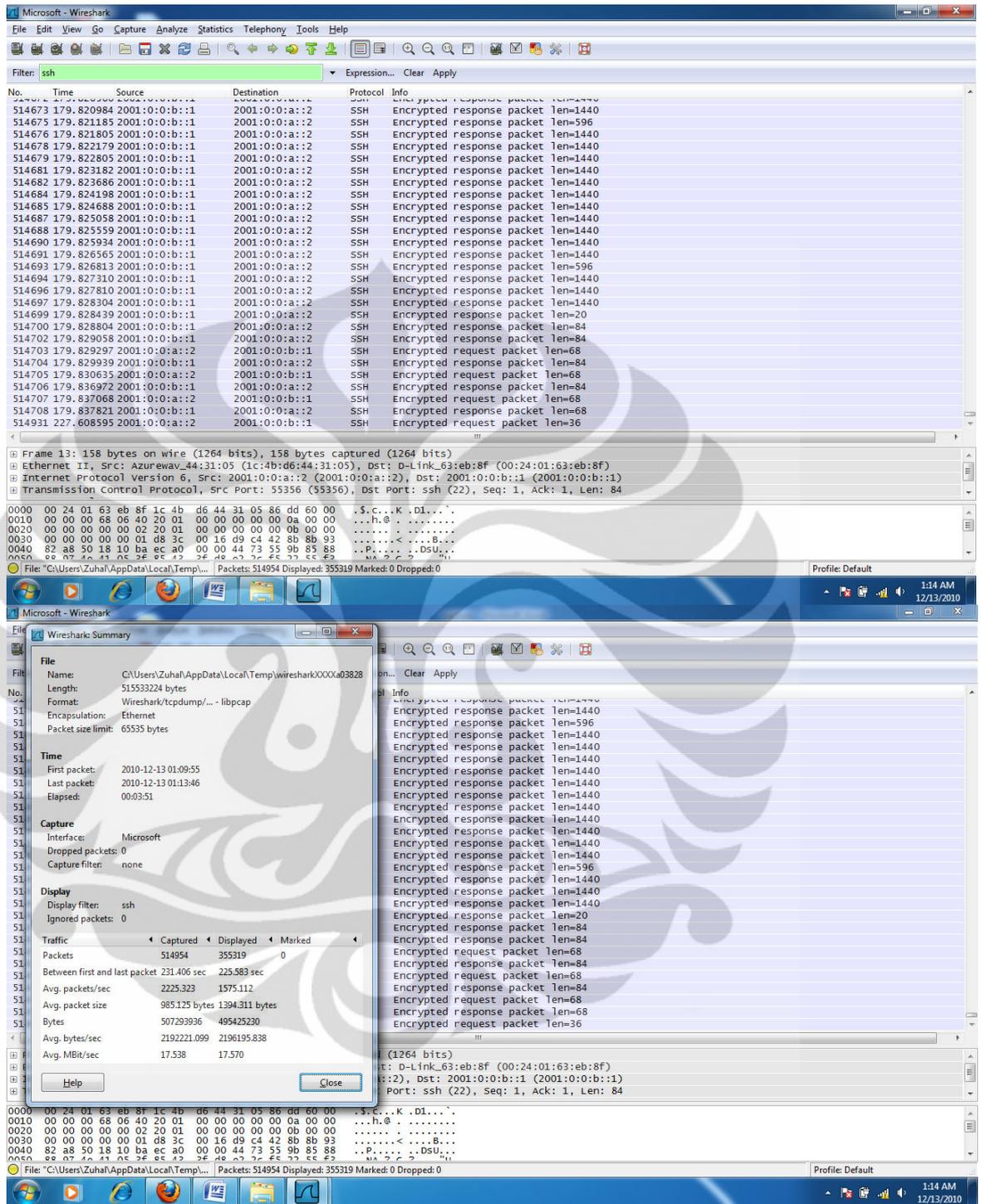
UPLOAD FILE 440 MIPv6



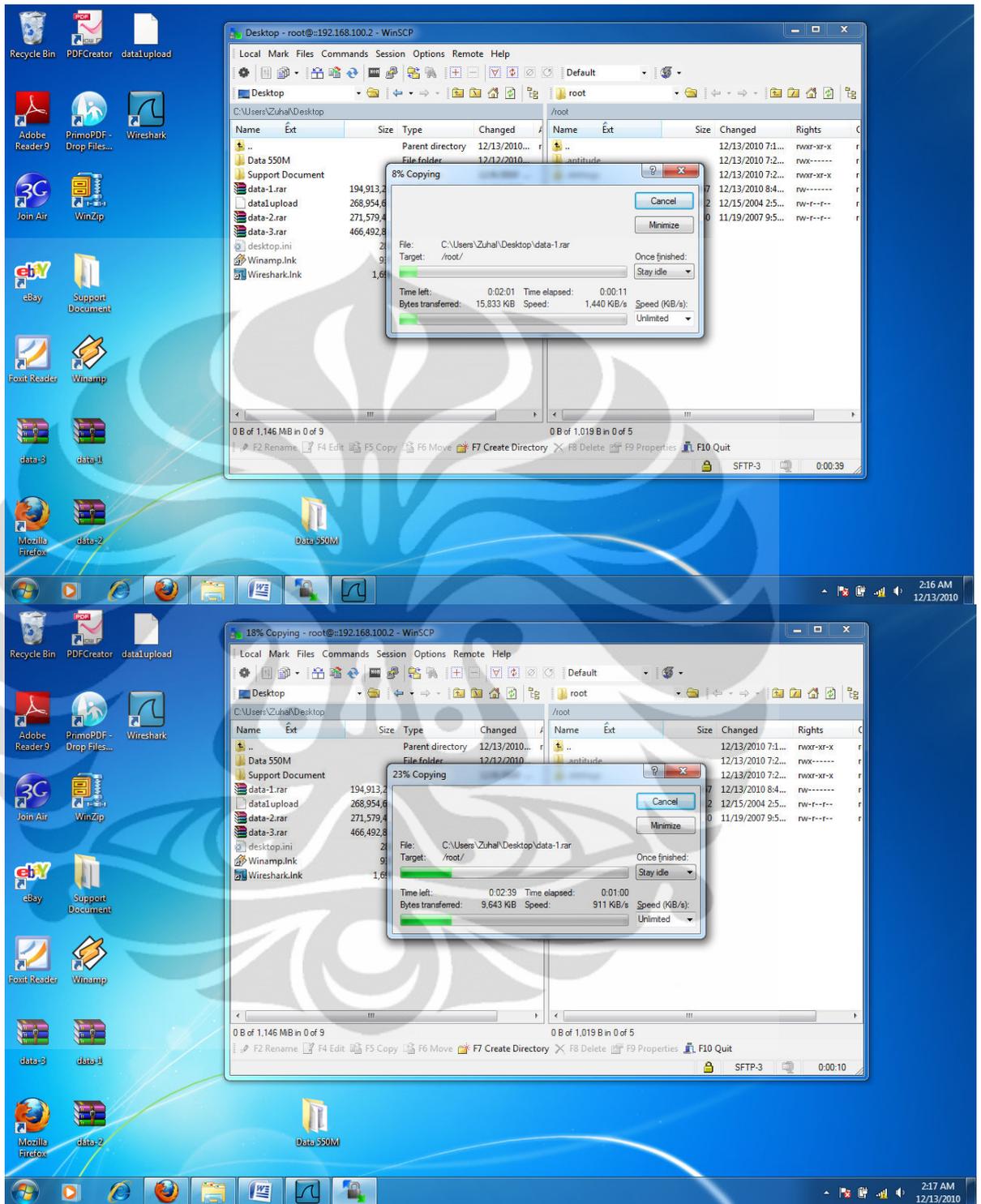


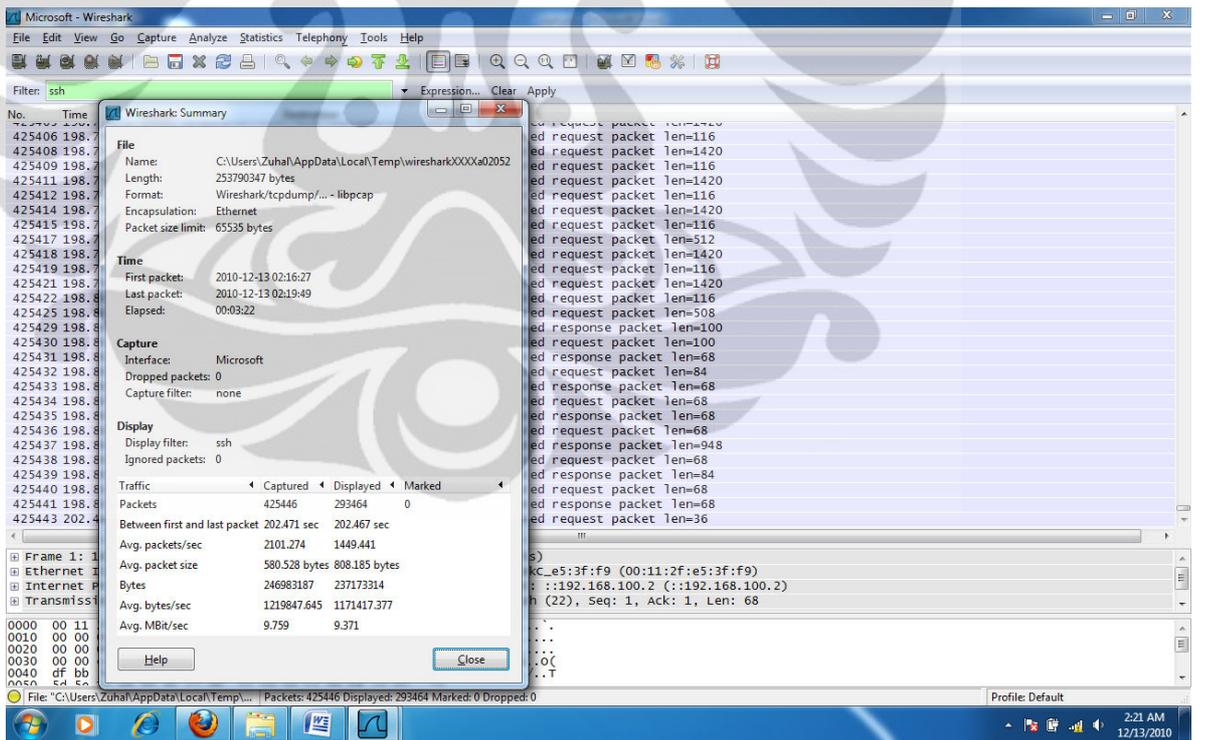
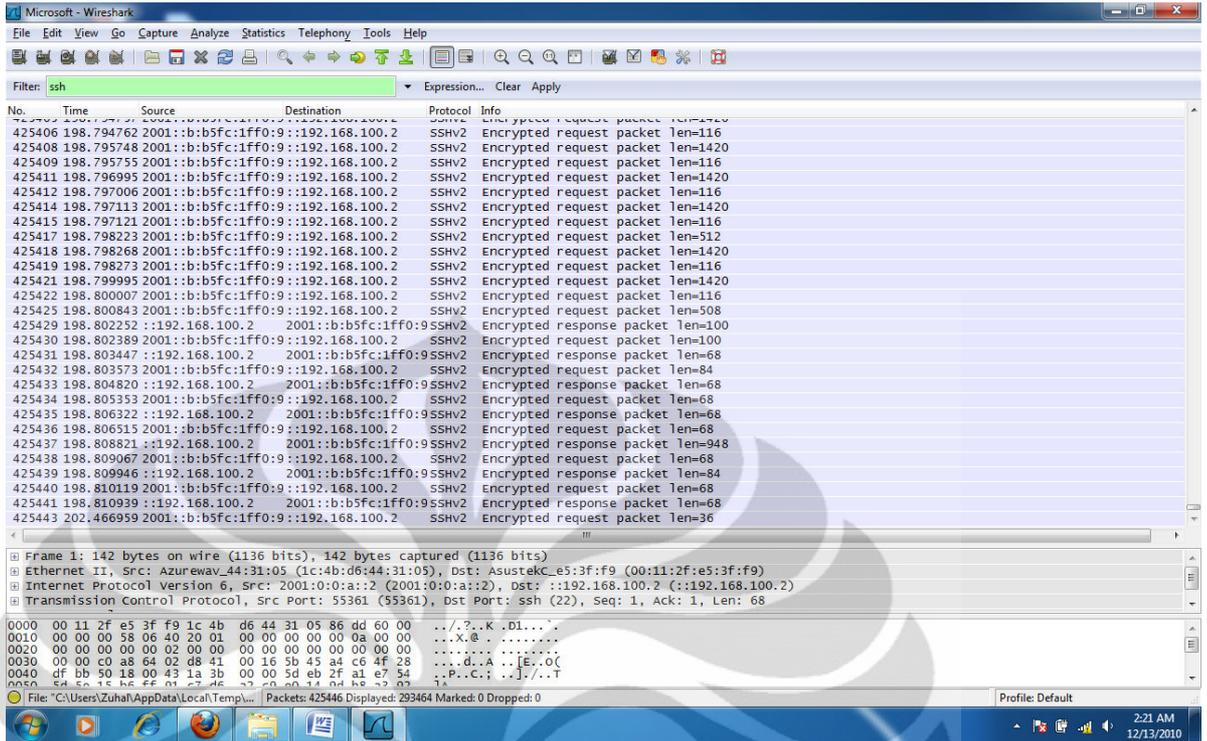
## DOWNLOAD FILE 440 MIPv6



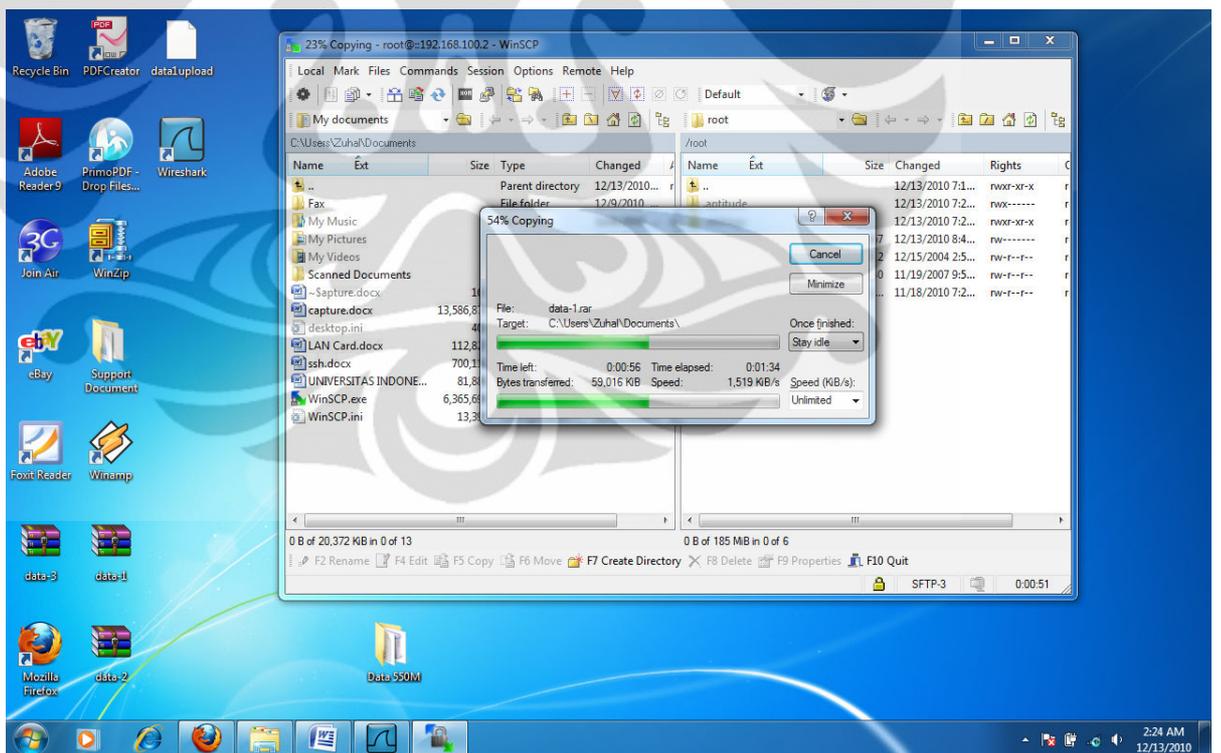
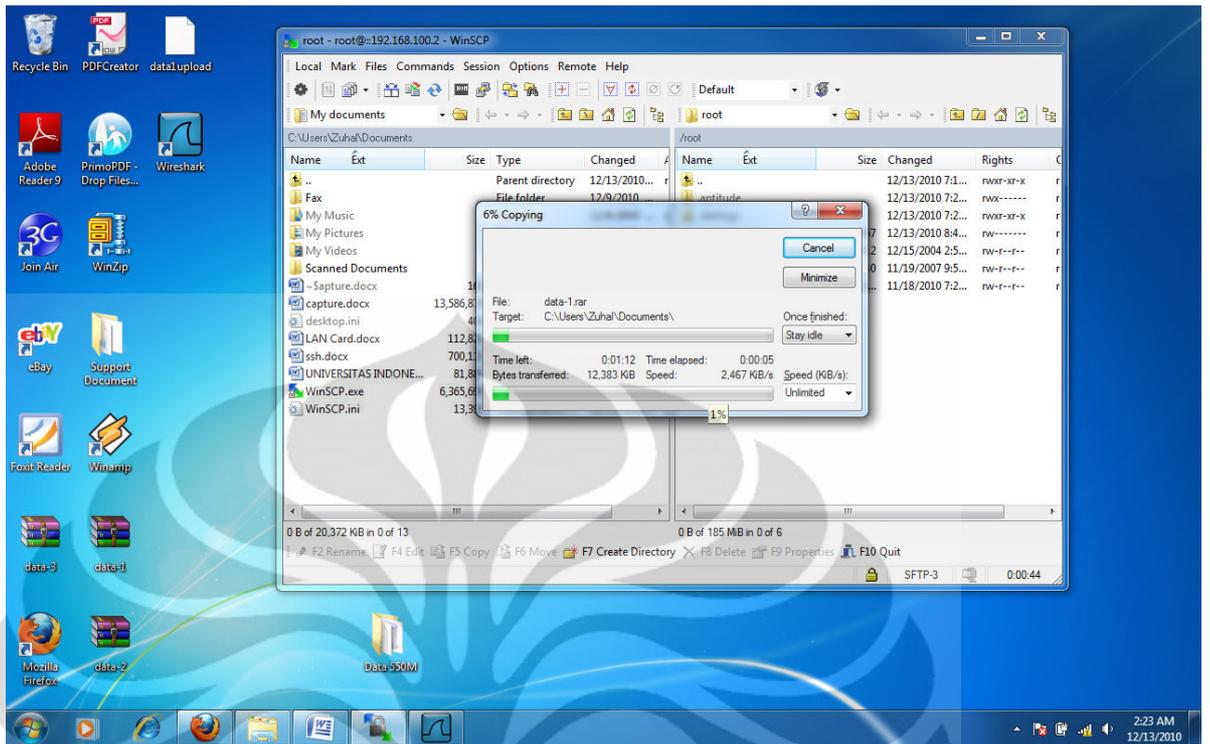


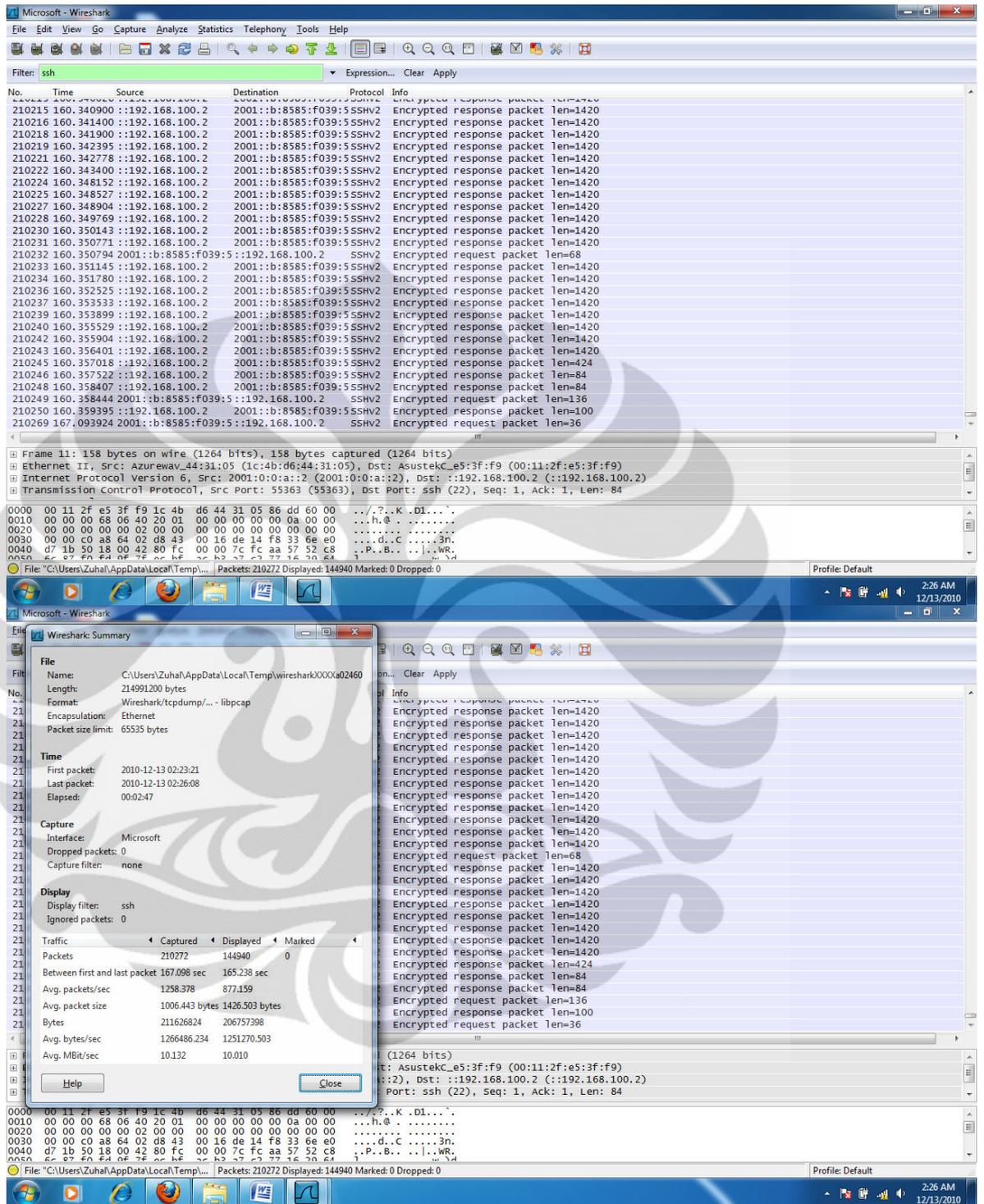
## UPLOAD FILE 190M MIPv6 TUNNELING 4to6



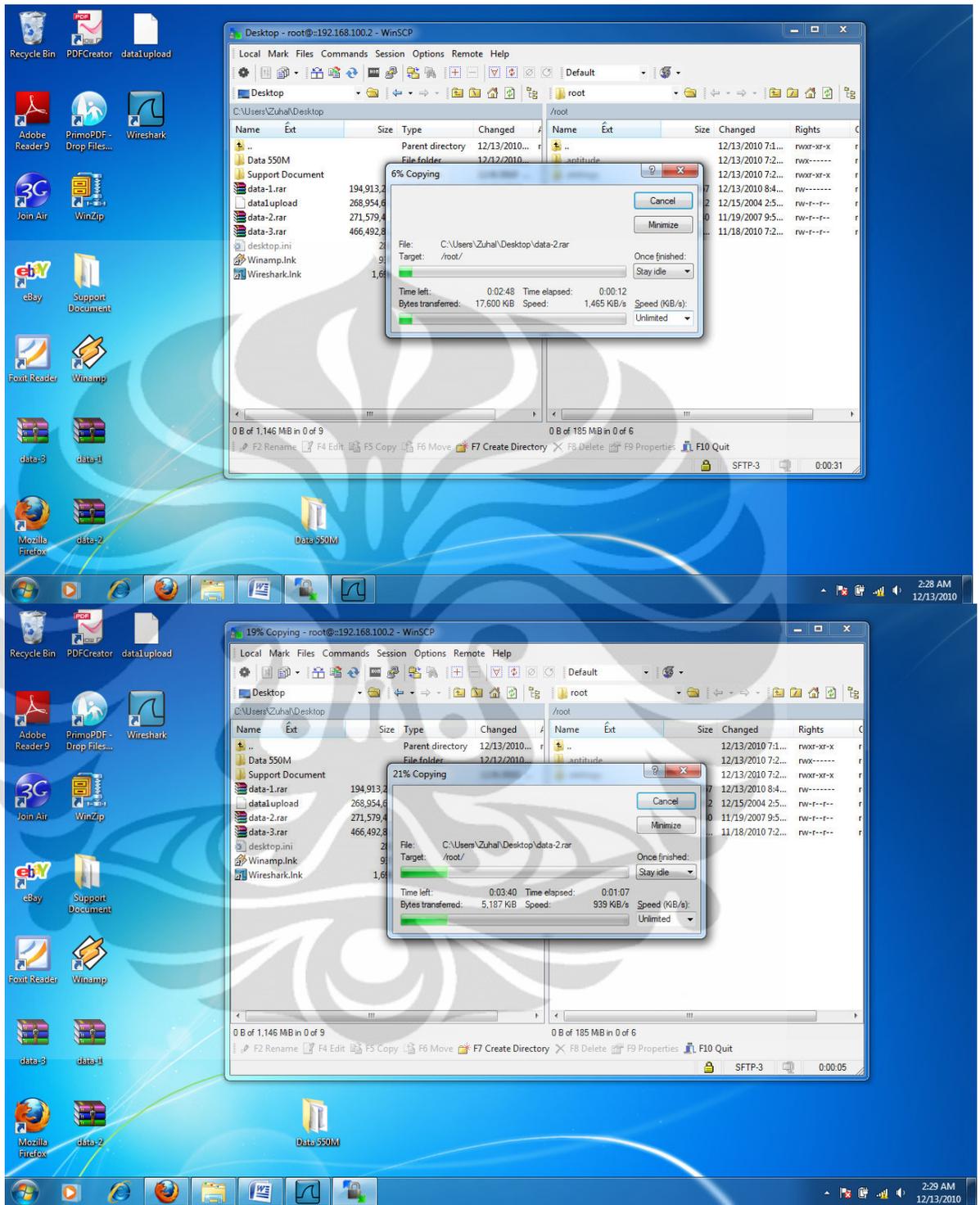


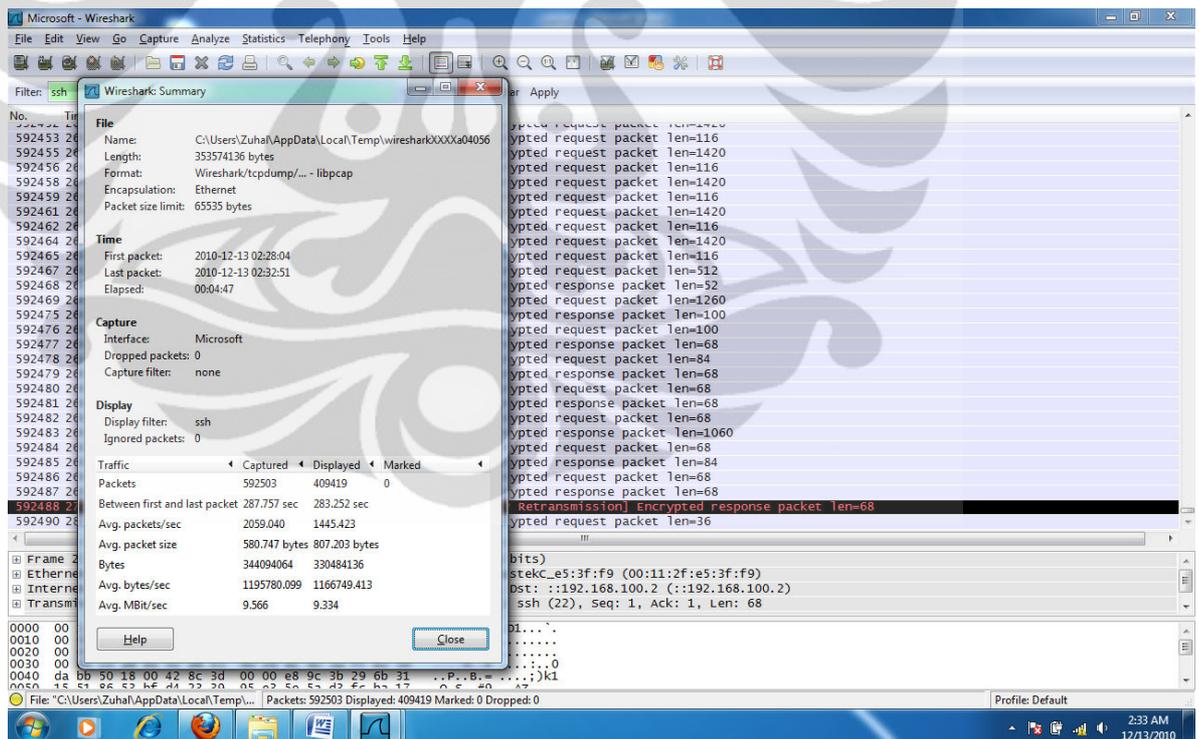
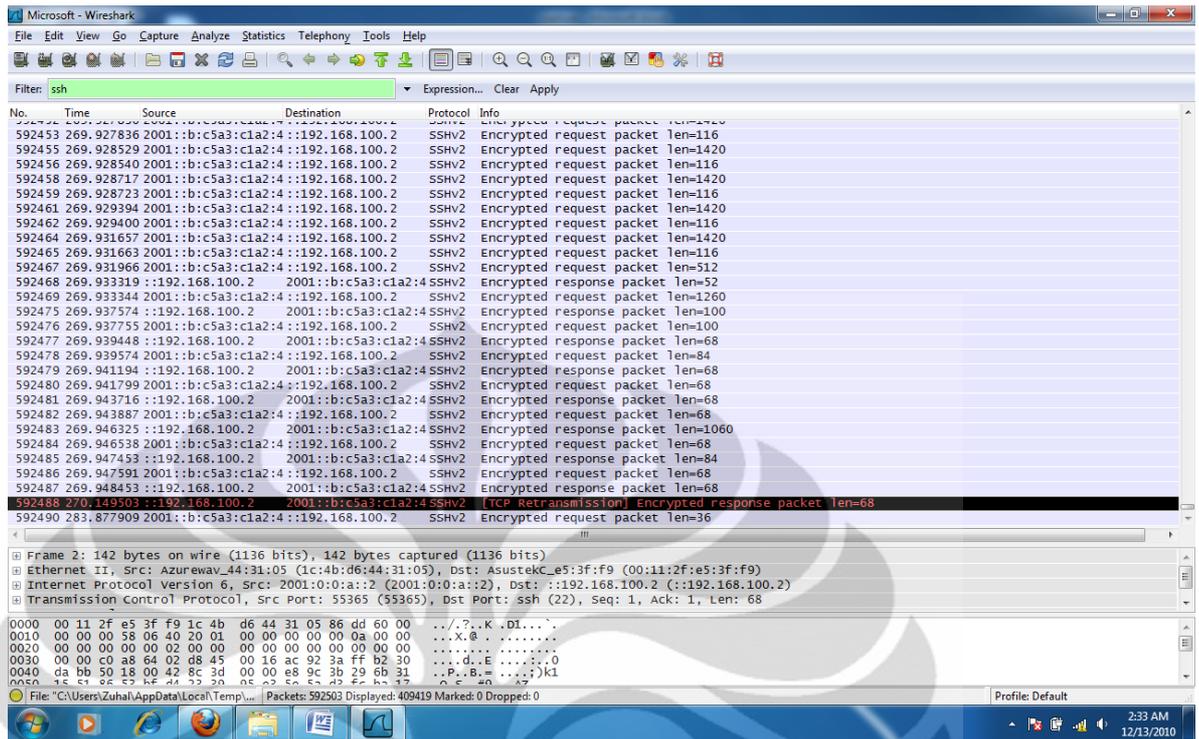
**DOWNLOAD FILE 160 MIPv6 TUNNELING 6to4**



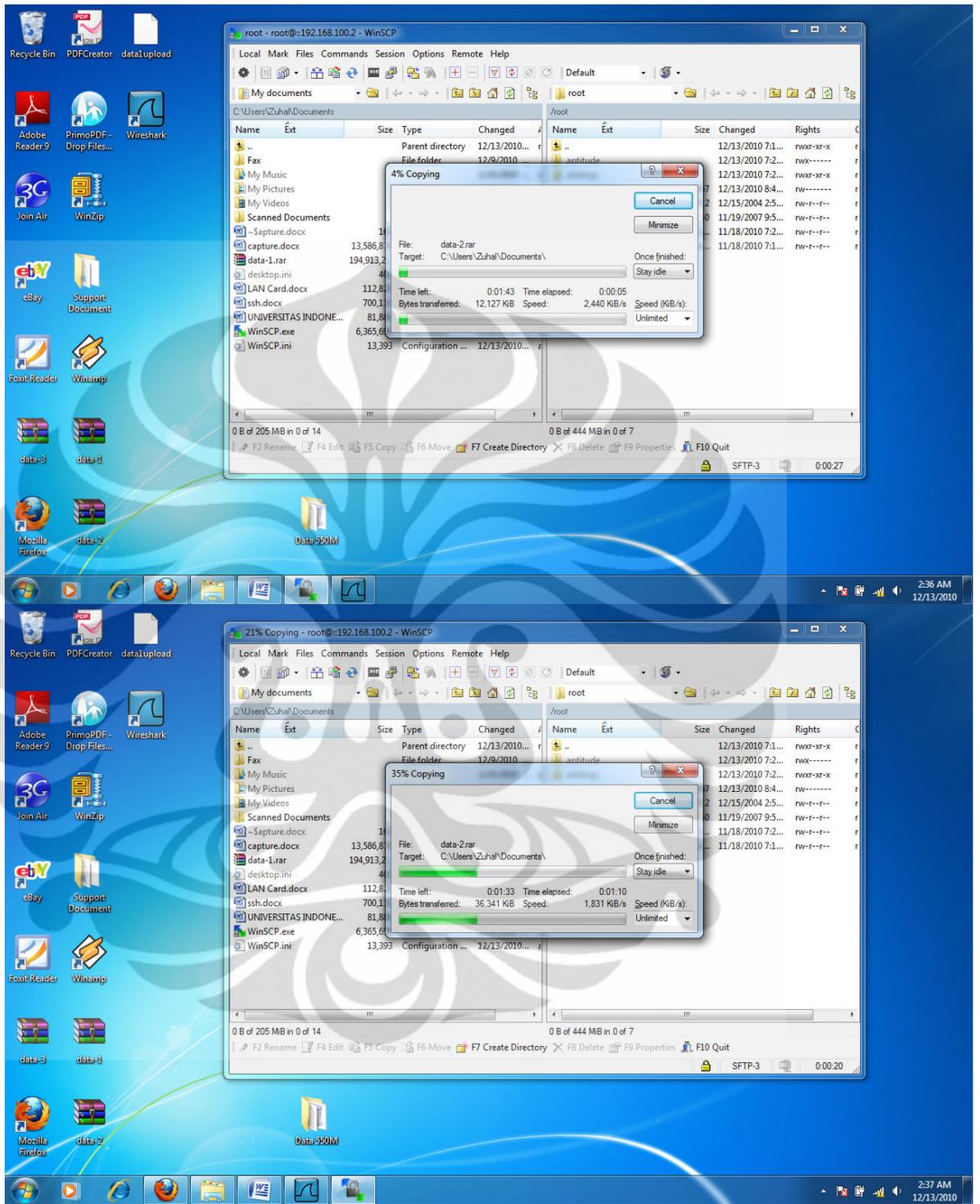


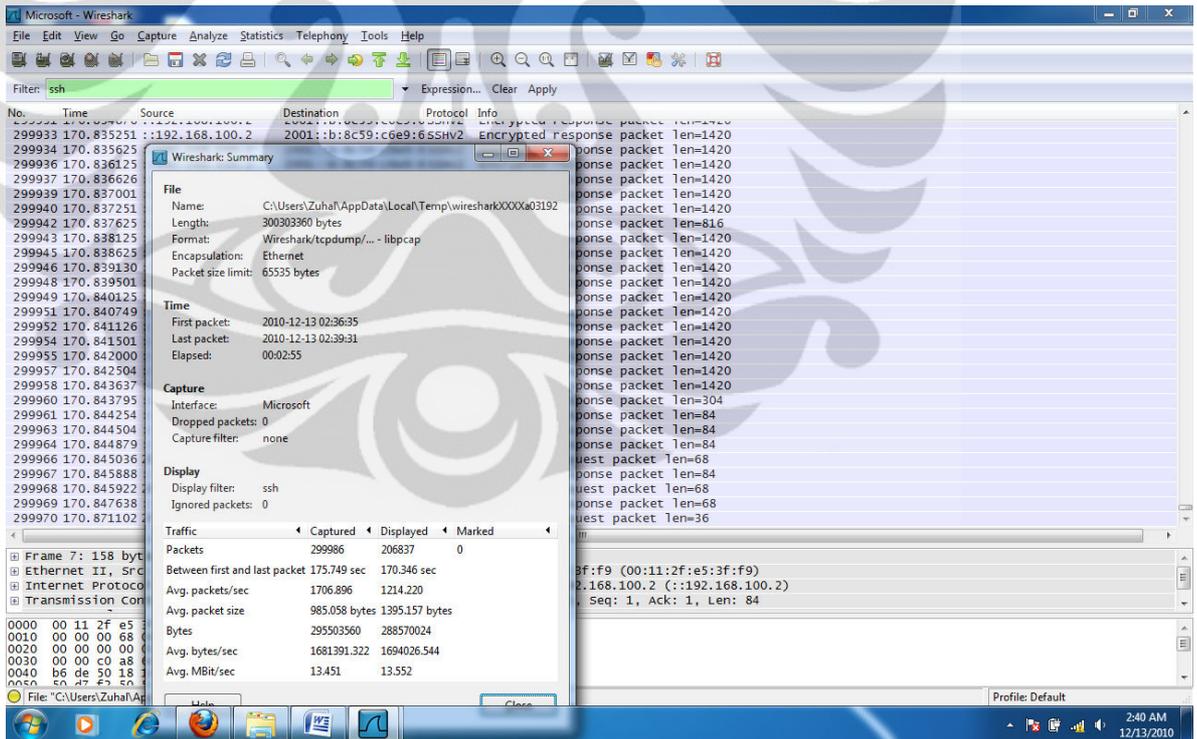
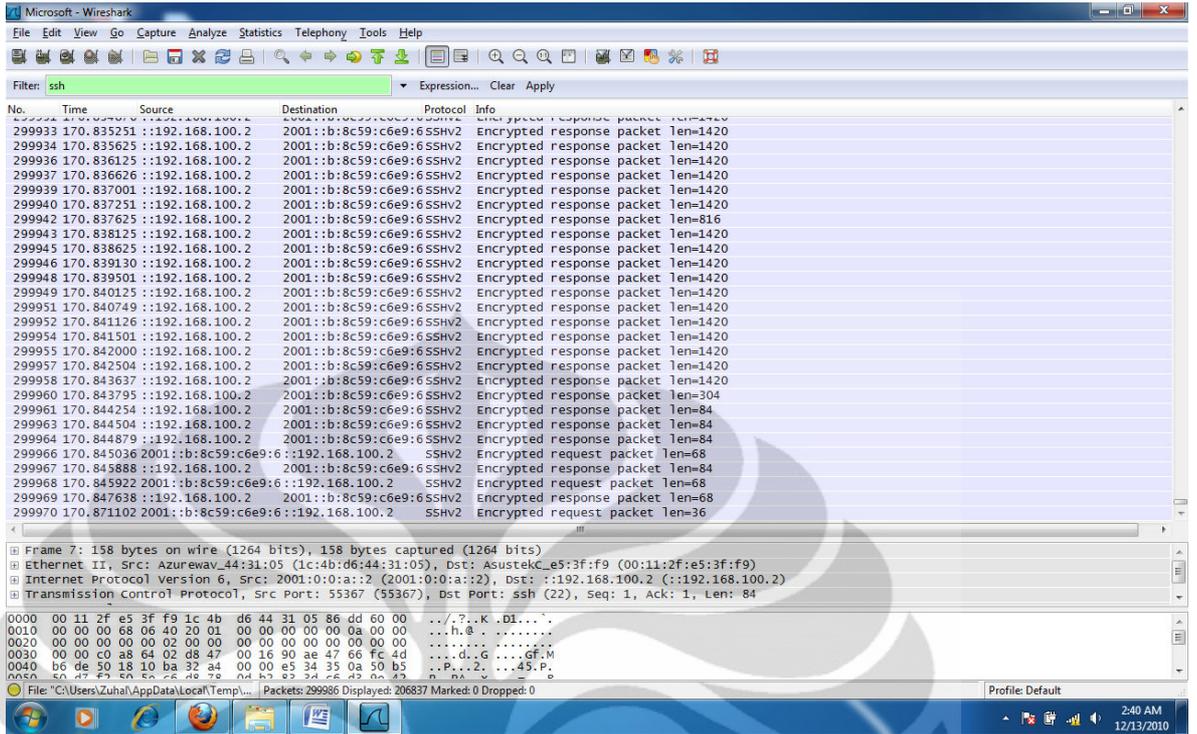
## UPLOAD FILE 260 MIPv6 TUNNELING 6to4



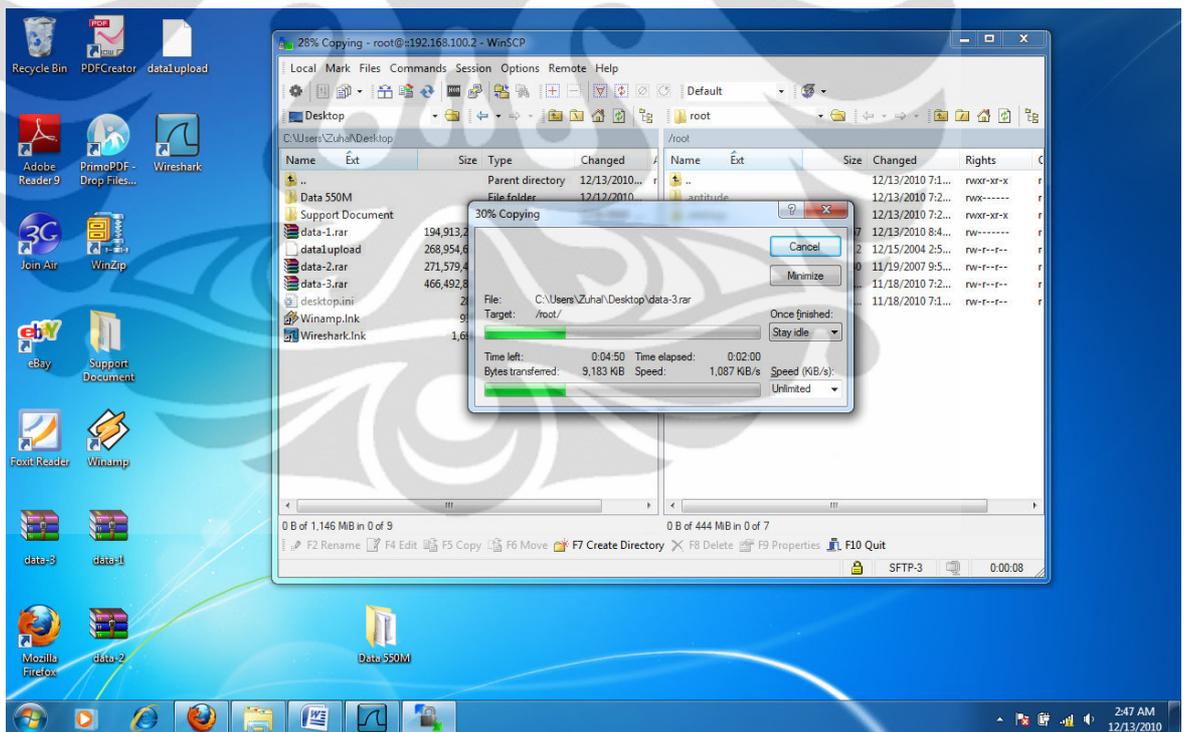
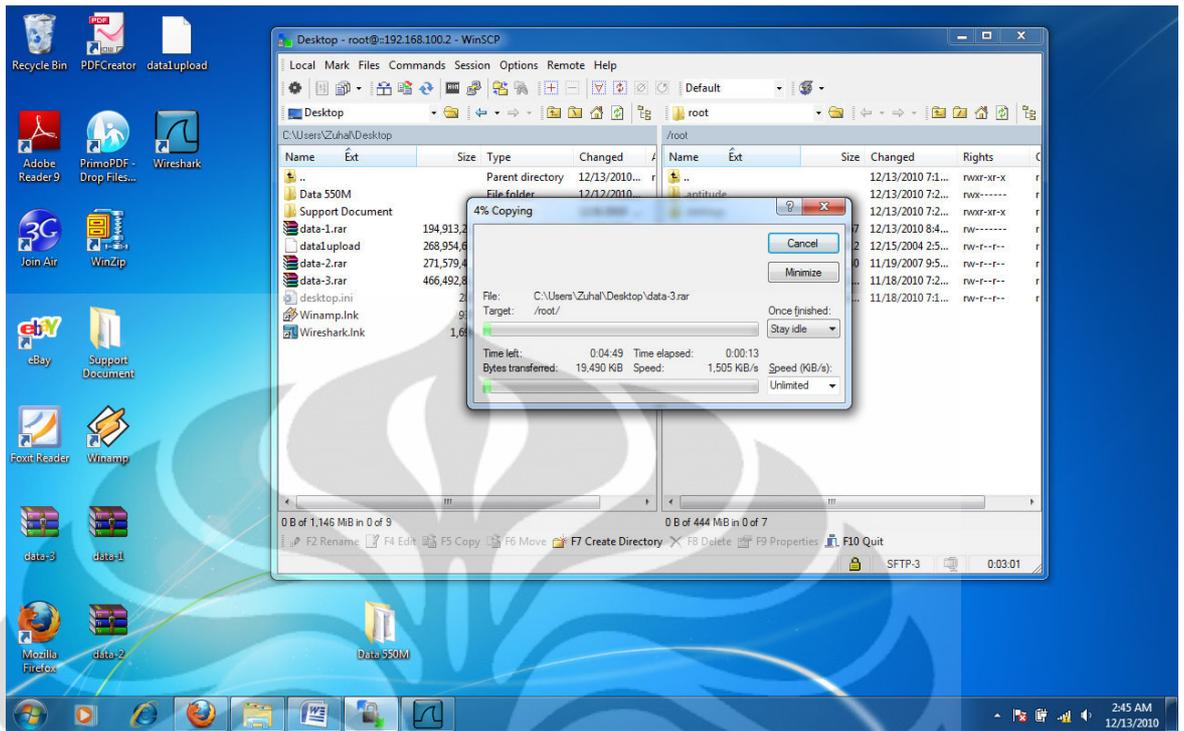


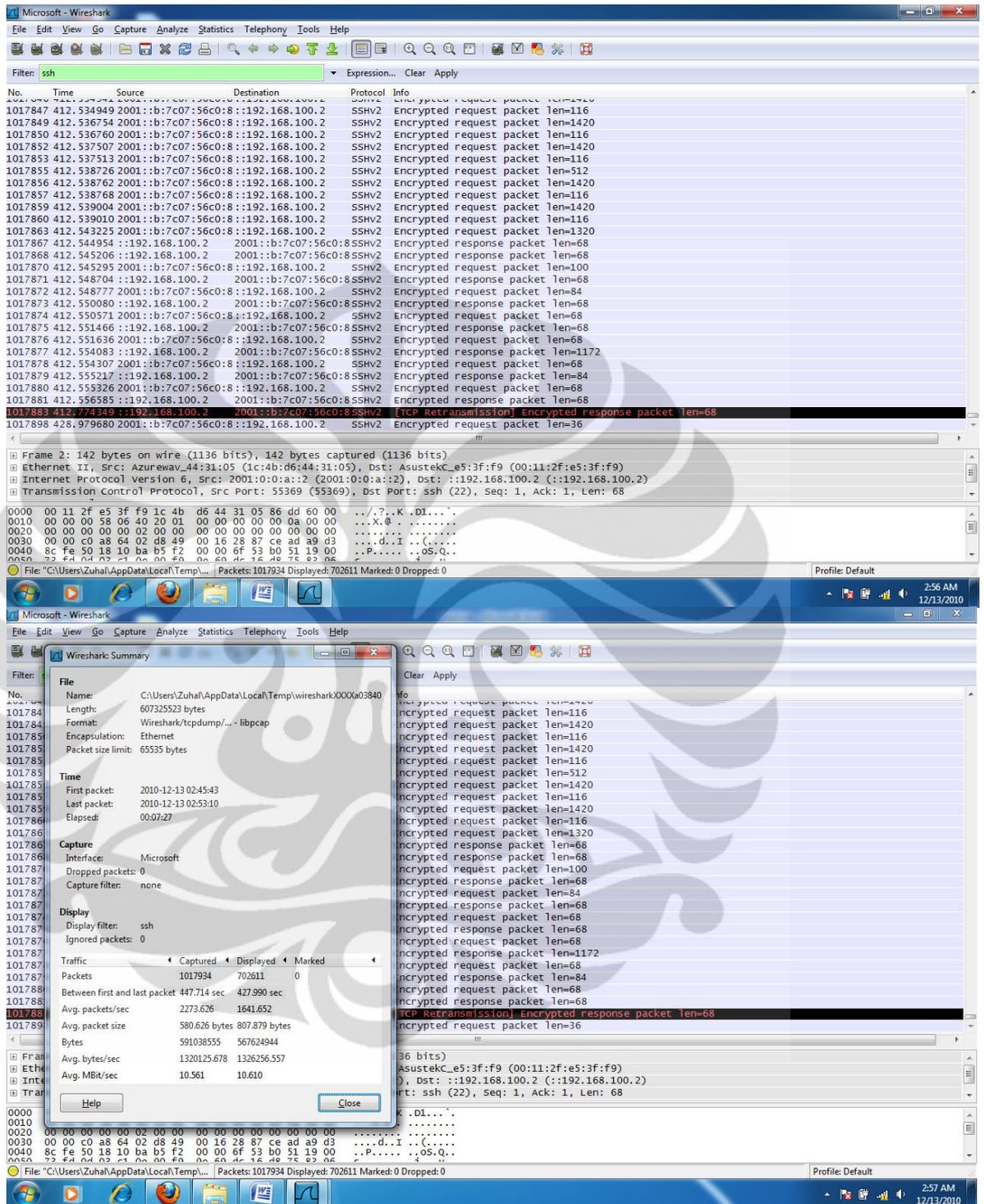
## DOWNLOAD FILE 260 MIPv6 TUNNELING 6to4



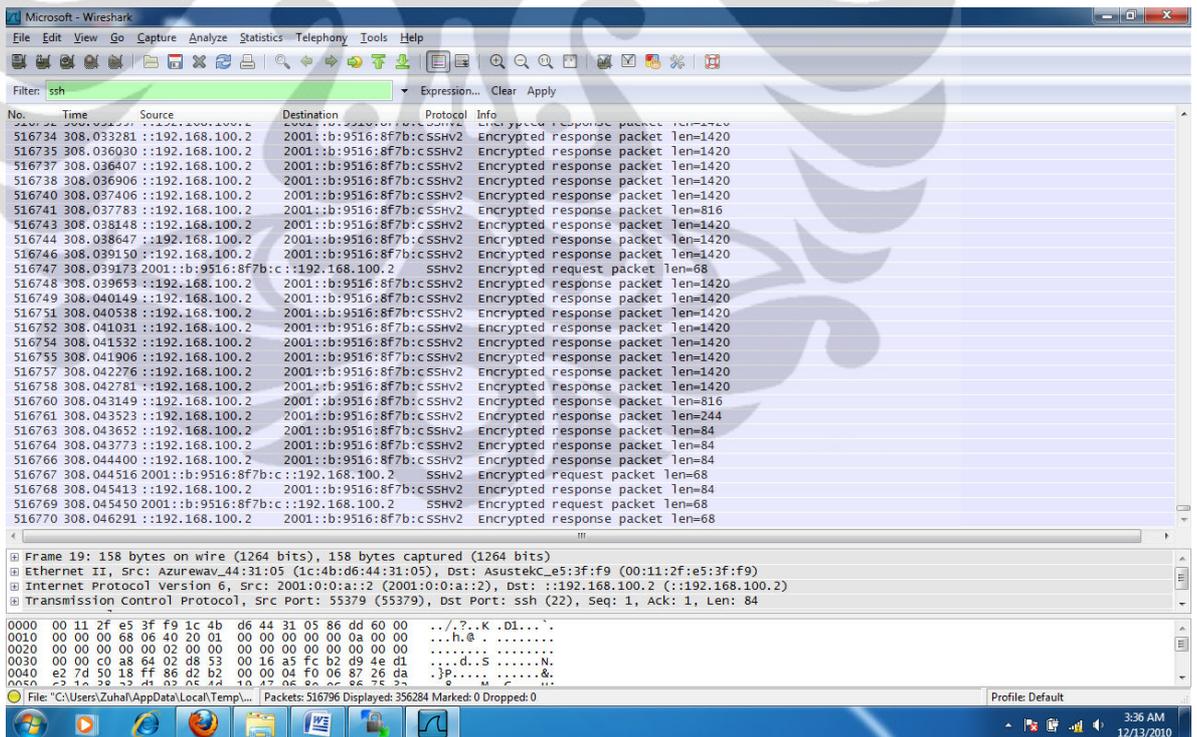
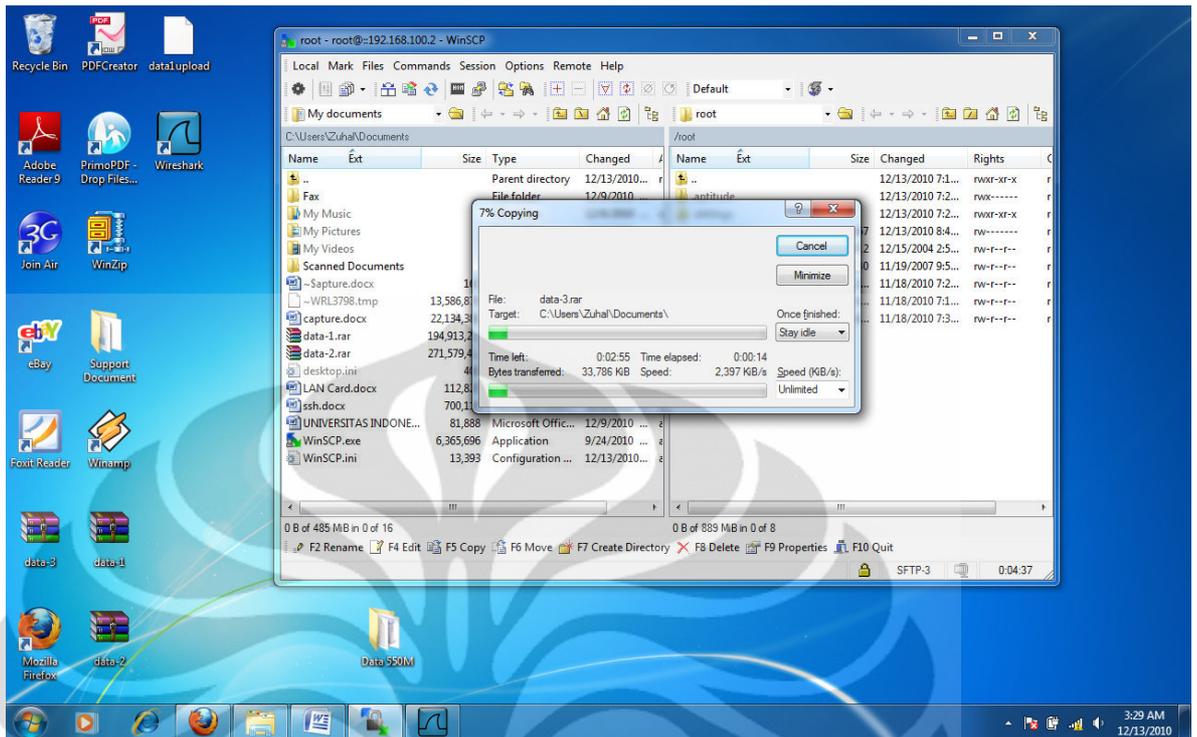


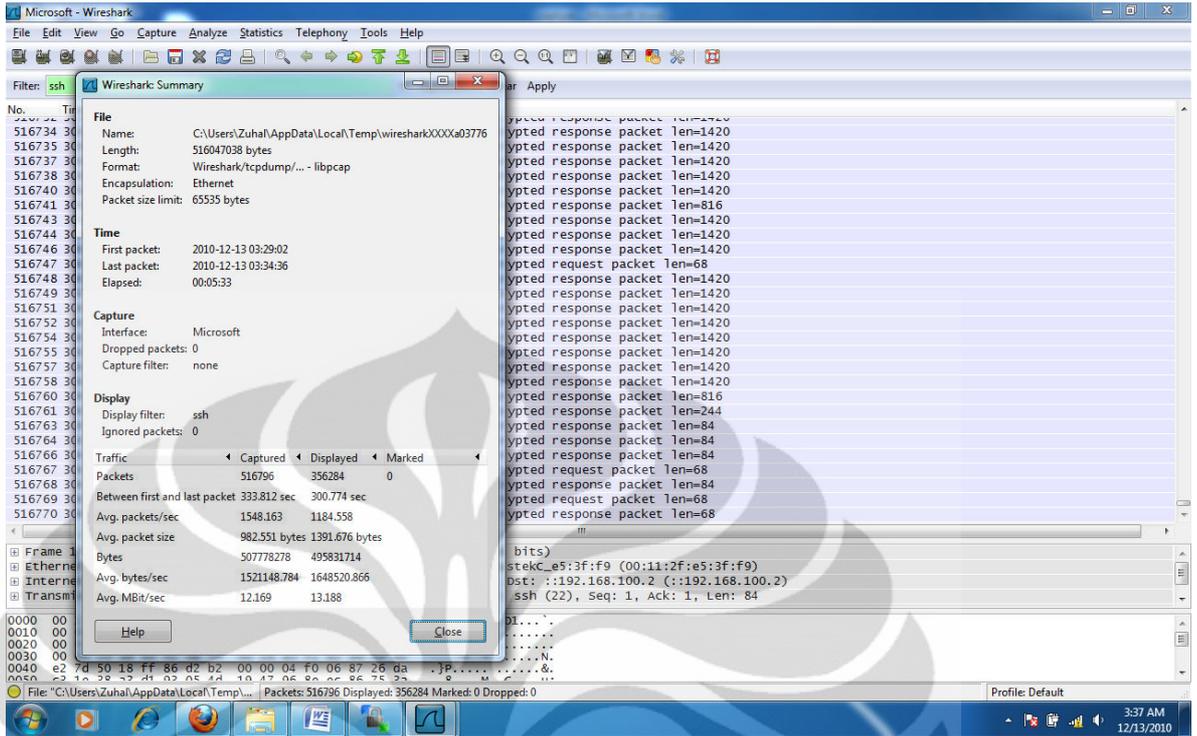
## UPLOAD FILE 440 MIPv6 TUNNELING 6to4





## DOWNLOAD FILE 440 MIPv6 TUNNELING 6to4





File(MByte)	Transfer time saat upload pada MIPv6 (detik)					
	1	2	3	4	5	Rata-rata
190	182,453	180,502	178,610	180,212	176,751	180,379
260	225,351	227,069	220,488	218,695	219,932	221,416
440	336,686	338,754	329,628	328,760	328,815	331,136

File(MByte)	Transfer time saat download pada MIPv6(detik)					
	1	2	3	4	5	Rata-rata
190	119,512	121,751	117,325	112,821	115,870	116,861
260	132,461	134,727	130,432	124,115	219,261	129,261
440	228,543	230,298	226,443	220,712	224,667	225,583

File(MByte)	Transfer time saat upload pada MIPv6 tunneling 6to4(detik)					
	1	2	3	4	5	Rata-rata
190	205,422	207,557	201,432	198,290	201,759	202,467
260	286,421	288,098	282,651	278,632	282,554	283,252
440	430,525	432,544	428,270	428,321	326,065	427,990

File(MByte)	Transfer time saat download pada MIPv6tunneling 6to4 (detik)					
	1	2	3	4	5	Rata-rata
190	168,254	170,323	166,096	160,880	164,088	165,238
260	173,445	175,679	170,425	165,980	169,310	170,346
440	303,675	305,427	301,721	295,005	300,091	300,774

File(MByte)	Throughput saat upload pada MIPv6 (Mbps)					
	1	2	3	4	5	Rata-rata
190	1,34	1,36	1,31	1,30	1,29	1,31
260	1,46	1,54	1,50	1,50	1,48	1,49
440	1,85	1,87	1,81	1,77	1,80	1,82

File(MByte)	Throughput saat download pada MIPv6 (Mbps)					
	1	2	3	4	5	Rata-rata
190	1,79	1,82	1,76	1,72	1,77	1,77
260	2,26	2,28	2,22	2,17	2,23	2,23
440	2,23	2,25	2,21	2,17	1,19	2,20

File(MByte)	Delay saat download pada MIPv6(mikro detik)					
	1	2	3	4	5	Rata-rata
190	0,570	0,568	0,565	0,560	0,562	0,565

260	0,457	0,452	0,450	0,467	0,465	0,458
440	0,605	0,598	0,604	0,603	0,605	0,606

File(MByte)	Delay saat upload pada MIPv6 tunneling 6to4(mikro detik)					
	1	2	3	4	5	Rata-rata
190	0,859	0,862	0,854	0,849	0,851	0,854
260	0,856	0,851	0,849	0,848	0,854	0,857
440	0,755	0,746	0,752	0,751	0,755	0,754

File(MByte)	Delay saat download pada MIPv6 tunneling 6to4 (mikro detik)					
	1	2	3	4	5	Rata-rata
190	0,803	1791	0,799	0,794	0,796	0.799
260	0,589	0,594	0,592	0,591	0,587	0,590
440	0,607	0,598	0,604	0,603	0,605	0,606