ANALISA IMPLEMENTASI INTERKONEKSI ANTARA IPv4 DENGAN IPv6 MENGGUNAKAN METODE *DUAL STACK* PADA MIKROTIK ROUTEROS (Studi Kasus : PT. Linggo Daya Energi)

Oleh:
Indra Warman, Mohd Yudhistira Septian Nugraha
Jurusan Teknik Informatika, Fakultas Teknologi Industri
Institut Teknologi Padang
Jl. Gajah Mada, Kandis Nanggalo, Padang
indrawmn@gmail.com, idmysn@gmail.com

Abstrak

Perkembangan jaringan komputer yang begitu pesat pada saat sekarang khususnya pada Internet Protokol. Pengguna Internet Protokol biasanya menggunakan IPv4 dan pengguna IPv4 saat ini tersisa 10%. Oleh karena itu diperlukan sebuah protokol yang bisa menampung Internet Protokol dalam skala yang besar yaitu IPv6 yang dapat menampung 2^{128} alamat IP. Pada Penelitian ini implementasi IPv6 menggunakan metode *Dual Stack* yang merupakan salah satu metode migrasi ke IPv6 yang dapat menjalankan dua IP secara bersama dalam satu *interface* yang sama dan menggunakan mekanisme migrasi 6to4, untuk pengujian pada Mikrotik RouterOs dengan melakukan *download* pada Xlight FTP Server. Dari hasil pegujian *Dual Stack* mengunakan mekanisme migrasi 6to4 tidak signifikan pengaruhnya pada pengujian *download file* dengan parameter *Quality of Services (QoS).*

Kata kunci: IPv4, IPv6, Migrasi, Mikrotik, *Dual Stack, Quality of Services (QoS)*

Abstract

The rapid development of computer networks at the present time especially in the internet protocol. Users of internet protocols typically use IPv4 and currently users IPv4 under 10%. Therefore required a protocol that can accommodate the internet protocol on a large scale that is IPv6 that can hold 2^{128} IP addresses. On the research IPv6 implementation can use the Dual Stack method which is one of the immigration methods to IPv6 which can run two IP together in the same interface and using the migration mechanism 6to4, for testing on Mikrotik RouterOs by downloading on Xlight FTP Server. From the result of testing the dual Stack using the migration mechanism 6to4 does not significantly affect the test download file with parameters Quality of Services (QoS).

Keywords :*IPv4, IPv6, *Migration, Mikrotik, Dual Stack , Quality Of Services (QoS)

1. Pendahuluan

Pada saat ini perkembangan teknologi semakin pesat, terutama pada teknologi komunikasi yang berbasis sistem pengalamatnya atau biasanya *Internet Protocol* (IP). Pada Internet Protokol Version (IPv4) yang hanya tersisa di bawah 10% pada saat ini. Oleh karena itu diperlukan sistem pengalamatan *Internet Protocol Version 6* (IPv6) yang dijadikan sebuah standarisasi baru pada sistem Pengalamatan yang lebih bisa mengakomendasi lebih banyak Pengalamatan.

IPv6 adalah sebuah standarisasi *Internet Protocol* terbaru yang dirancang pada tahun 1994. IPv6 memiliki kombinasi untuk pengalamatan 2^{128} , cukup memberikan setiap orang di dunia IP dengan kode unik. Dalam IPv6 ini, 128-*bit* untuk di bagi dalam 8 blok berukuran 16-*bit* setiap bloknya dan dapat dikonversikan dalam bentuk hexadesimal berukuran 4-*bit* setiap blok bilangan hexadesimal tersebut terpisah dengan titik dua. Berbeda dengan IPv4 yang berbasis 32-*bit*,yang memiki kapasitas 2^{32} atau 4,295 x 10^9 alamat sedangkan IPv6 yang berbasis 128-*bit*, yang memiliki kapasitas 2^{128} atau sekitar 3 x 10^{38}alamat.

Penggunaan IPv6 di Indonesia hanya berkisaran kurang dari 1% dari pemakaian IPv6 di seluruh dunia. Penggunaan IPv6 di

DOI 10.21063/JTIF.2017.V5.2.63-72

Indonesia untuk sekarang ini di kalangan akademisi dan riset contohnya Institut Teknologi Bandung (ITB). Penggunaan IPv6 di seluruh dunia hanya sekitar 16% dari pengunaan *Internet Protokol* (IP) di seluruh dunia.

IPv6 mempunyai *header* yang dan format alamat yang sangat berbeda dibandingkan IPv4, sehingga IPv4 tidak dapat berhubung langsung dengan IPv6 dan masih banyak *Internet Service Provider* (ISP) mengunakan IPv4. Hal Ini akan akan menimbulkan masalah pada sisi pengimplementasi IPv6. Oleh karena itu perlu adanya transisi untuk mengatasi permasalahan antara interkoneksi antara IPv6 dengan IPv4. Metode migrasi mengunakan *Dual Stack* bisa mengatasi permasalahan Implementasi migrasi antara IPv6 dengan IPv4.

Dual Stack Adalah Metode migrasi untuk menjalakan proses implementasi interkoneksi antara memungkinkan satu *Interface* dapat menggunakan IPv4 dan IPv6 secara bersamaan dan karena IPv6 ini masih sedikit yang mengunakan untuk satu jenis *Internet Protokol version 6* saja atau IPv6 *only*. Maka dengan menggunakan metode *Dual Stack* ini bisa mengatasi permasalahan migrasi yang muncul selama ini didalam pengimplementasian IPv6.

Berdasarkan dari permasalahanan diatas, dengan metode transisi *Dual* Stack untuk migrasinya antara IPv4 dengan IPv6, Berdasarkan hal tersebut diatas dapat dirumuskan dalam bentuk penelitian dengan judul "Analisa Implementasi Interkoneksi Antara IPv4 Dengan IPv6 Menggunakan Metode *DualStack* Pada Mikrotik Router OS®".

2. Metodologi

Jenis penelitian ini merupakan penelitian eksperimen, yaitu menganalisa kinerja *Internet Protokol Version 6* (IPv6) menggunakan metode *Dual Stack* untuk migarasi jaringan komputer dengan menggunakan dua *internet protokol* yang berbeda yaitu IPv4 dan IPv6 dengan satu *interface* pada Mikrotik *Router*Os. Parameter yang diuji yaitu *Transfer Time, Delay* dan *Throughput*. Parameter yang diuji menggunakan akses melalui jaringan LAN dan untuk datanya diambil dengan meng-*download* dari server

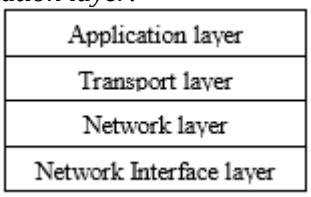
2.1 Jaringan Komputer

Jaringan komputer adalah sekumpulan komputer otonom yang terinterkoneksi oleh teknologi yang sama dan dapat saling bertukar informasi. Koneksi dapat menggunakan media kawat tembaga, serat optik, gelombang mikro, sinar inframerah ataupun menggunakan komunikasi satelit

2.2 TCP/IP

TCP dan IP merupakan salah satu standar protokol yang dirancang untuk melakukan fungsi-fungsi komunikasi data dalam jaringan internet. TCP/IP terdiri atas sekumpulan protokol yang masing-masing bertanggung jawab atas bagian-bagian tertentu dalam komunikasi data. Dengan prinsip ini maka tugas masing-masing protokol menjadi jelas dan sederhana, sehingga mudah untuk diimplementasikan di seluruh perangkat dan perangkat lunak jaringan dan juga mudah dalam melakukan proses *troubleshooting*.

Arsitektur TCP/IP dapat dimodelkan dalam empat lapisan TCP/IP, yaitu *network interface layer*, *network layer*, *transport layer* dan *application layer*.



Gambar 1 Arsitektur TCP/IP

Dalam proses pengiriman data antar layer, setiap layer akan menganggap informasi yang datang dari layer sebelumnya sebagai data, sehingga ia akan menambahkan informasi miliknya pada data tersebut. Begitu juga sebaliknya, jika ia menerima data yang dianggap valid maka ia akan melepas informasi tersebut.

Network interface layer merupakan lapisan terbawah yang bertanggung jawab untuk mengirim dan menerima data ke dan dari media fisik. Oleh karena protokol dalam layer ini harus mampu merubah bit-bit informasi menjadi sinyal listrik.

Internet layer merupakan protokol yang bertanggung jawab dalam proses pengiriman paket ke alamat yang tepat dan bersifat *unreliable* dan *connectionless*. Pada layer ini terdapat tiga macam protokol yaitu

DOI 10.21063/JTIF.2017.V5.2.63-72

IP, ARP dan ICMP, *Internet Protocol* berfungsi untuk menyampaikan paket data ke alamat yang tepat. ARP (*Address Resolution Protocol*) ialah protokol yang digunakan untuk menemukan alamat hardware dari LAN card, dan Internet Control Massage Control berfungi sebagai memberikan kiriman pesan-pesan ke dalam sebuah jaringan, mulai dari mengirimkan pesan eror, pesan diterima.

Transport layer merupakan protokol yang bertugas untuk mengadakan hubungan dan mengatur transportasi data antara dua host/komputer. Protokol dalam lapisan ini, yaitu TCP dan UDP. TCP (*Transmission Control Protocol*) bersifat *reliable* dan *connection oriented*, Sedangkan UDP (*Unit Datagram Protocol*) bersifat *connectionless* dan *unreliable*

Application layer, merupakan lapisan teratas yang berisi semua aplikasi berbasis TCP & IP dan berhubungan langsung dengan pemakai. Aplikasi tersebut misalnya FTP, HTTP dan Telnet FTP (*File Transfer Protokol*) adalah program aplikasi untuk mentransfer file antara clien &*server*. HTTP (*Hyper Text Transfer Protocol*) adalah program aplikasi yang digunakan untuk menterjemahkan alamat IP menjadi susunan huruf yang dipisahkan dengan tanda "." misal http://www.itp.ac.id.

Dari beberapa macam protokol yang ada dalam TCP & IP, protokol IP merupakan inti dari protokol TCP & IP. Seluruh data yang berasal dari lapisan diatas IP harus dilewatkan, diolah oleh protokol IP dan kemudian dikirimkan sebagai paket IP ke tujuan. Dalam melakukan pengiriman paket, protokol IP bersifat *unreliable*, *connectionless* dan *datagramdelivery service*. Saat ini terdapat dua versi dari protokol IPv4 (32 bit) dan IPv6 (128 bit). *Unreliable* berarti protokol IP tidak menjamin datagram yang dikirim pasti sampai di tujuan. Protokol IP hanya berusaha sebaik mungkin untuk membawa *datagram* sampai ke tujuan. *Connectionles*s berarti dalam mengirim paket ke tujuan tidak ada perjanjian terlebih dahulu (*handshake*). *Datagramdelivery service* berarti paket data yang dikirim independent terhadap paket data yang lain. Akibatnya jalur yang ditempuh oleh masing-masing paket berbeda satu dengan lainnya.

2.3 Internet Protokol version 4 (IPv4)

Model pengalamatan dalam IPv4 menggunakan 32 bit bilangan biner. Namun untuk mempermudah penulisannya maka setiap delapan bit biner diwakili oleh satu segmen bilangan oktet, sehingga setiap alamat akan memiliki empat buah segmen dari 0.0.0.0 sampai dengan 255.255.255.255 misalnya 202.152.254.254 sehingga total alamat sebesar 2^{32}.

Alamat IPv4 dibagi menjadi dua bagian yaitu alamat jaringan (network *address*) dan alamat komputer (host *address*). Network *address* digunakan untuk menunjukkan di jaringan mana komputer berada, sedangakan "host *address*" menunjukkan komputer tersebut dalam jaringannya tersebut.

Untuk meningkatkan efisiensi dan mempermudah administrasi jaringan, maka dalam suatu jaringan yang besar perlu dibagi-bagi ke dalam jaringan yang lebih kecil. Konsep ini sering disebut dengan *subnetwork* / *subnetting.*

2.4 Internet Protokol version 6 (IPv6)

Standar IP yang saat ini digunakan pada kebanyakan jaringan adalah Internet Protokol Versi 4 (IPv4). IPv4 dikembangkan pada awal tahun 1970 untuk memfasilitasi komunikasi dan pertukaran informasi antara peneliti di bidang pemerintahan dan bidang akademik. Pada waktu itu sistem yang ada terbatas, sehingga pengembang IPv4 tidak terlalu mementingkan variabel security dan QoS. Jumlah alamat yang tersedia pada saat itu juga dirasa sangat mencukupi yang mencapai 2^{32} alamat.Saat ini jaringan komputer telah berkembang sangat pesat. Jumlah alamat pada IPv4 sudah tidak lagi mencukupi untuk memenuhi kebutuhan jaringan jaringan baru. Dukungan *security* dan QoS yang terintegrasi juga sangat dibutuhkan dalam kebanyakan konfigurasi jaringan dewasa ini. Untuk mengatasi kekurangan ini, IETF (*Internet Engineering Task Force*) pada tahun 1990 mulai mengembangkan Internet Protokol generasi baru yang dinamakan Internet Protokol Versi 6 (IPv6) yang mampu menampung jumlah alamat dengan 2^{128}alamat.

2.5 Format Pengalamatan IPv6

Berikut ini adalah contoh alamat IPv6 dalam bentuk bilangan biner yang

DOI 10.21063/JTIF.2017.V5.2.63-72

sudah di bagi kedalam 8 blok yang berukuran 16-bit setiap bloknya untuk menerjemahkan kedalam notasi *colon-hexadecimal format* adalah sebagai berikut :
0010000111011010 0000000011010011 0000000000000000 0010111100111011 0000001010101010000000011111111 1111111000101000 1001110001011010. Sesudah itu setiap blok yang berukuran 16-bit tersebut harus dikonversikan kedalam bilangan hexadecimal tersebut dipisahkan dengan titik dua hasilnya adalah seperti dibawah ini

21DA:00D3:0000:2F3B:02AA:00FF:FE28:9C5A

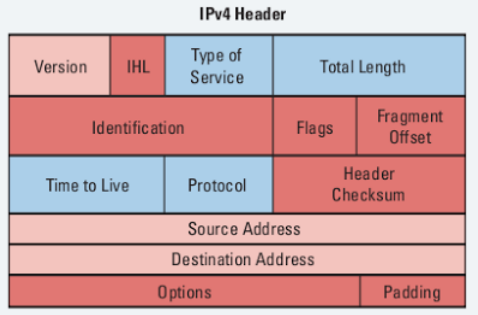
Alamat diatas juga dapat disederhanakan lagi dengan membuangkan angka 0 pada awal setiap blok yang berukuran dengan menyisakan satu digit terakhir. Dengan membuang 0 alamat diatas menjadi :

21DA:D3:0:2F3B:2AA:FF:FE28:9C5A.

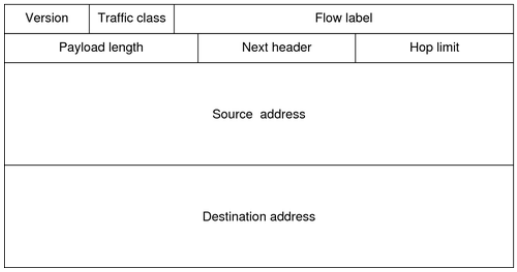
Konvensi pengalamatan juga IPv6 juhga mengizinkan penyederhanaan alamat lebih jauh lagi, yakni dengan membuang karakter 0, pada sebuah alamat yang banyak angka 0-nya. Jika sebuah alamat IPv6 direpresentasikan dalam notasi *colon-hexadecimal format* mengandung nenerapa blok dengan angka o, maka alamat dapat disederhanakan dengan mengunakan tanda dua buah titik dua (::). Penyerhanaan alamat IPv6 kali ini sebaiknya digunakan sekali didalam satu alamat, karena kemungkinan nantinya pengunaan tidak dapat memtukan beberapa banyak bit 0 yang direpresentasikan oleh setiap titik dua titik dua (::) yang terdapat dalam alaat tersebut. Beikut adalah alamat yang disederhanakan dua titik dua (::). Berikut ini tabel 2.2 yang mengilustrasikan penyerderhanaan alamat pada IPv6.

2.6 Format Header IPv4 dan IPv6

Format IPv6 memiliki format *header* sederhana dibandingkan dengan format header format IPv4, diketahui bahwa header IPv6 memiki ukuran yang sebesar 40 byte. Dua field source dan destination masing masing menggunakan 16 byte(128 bit), sehingga tersisa 8 byte untuk keperluann informasi header umum berikut ini perbedaan header IPv4 dengan IPv6 :



Gambar 2 Header IPv4



Gambar 3 Header IPv6

2.7 Mekanisme Transisi

Mayoritas penggunaan teknologi internet pada saat ini memakai protokol IPv4, sehingga infrastruktur yang digunakan sekarang memiliki kendala untuk melakukan interkoneksi atau transisi protokol dari IPv4 ke IPv6. Sebagai lembaga yang mengatur masalah tersebut, IETF (*Internet Engineering Task Force)* telah membentuk tim untuk mengatasi proses Interkoneksi dari IPv4 ke IPv6 yaitu IETF IPng *Transition*. Tujuan pembuatan mekanisme transisi ini supaya paket IPv6 dapat dilewatkan pada jaringan IPv4 yang telah ada ataupun sebaliknya. Proses Interkoneksi dari IPv4 ke IPv6 dilakukan secara bertahap sehingga tidak mengganggu jaringan yang telah berjalan sebelumnya. Selama proses Interkoneksi jaringan IPv4 dan IPv6 harus dapat saling berkomunikasi tanpa mengurangi kehandalannya.

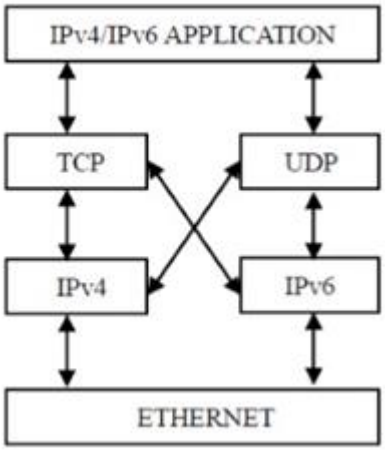
Mekanisme transisi secara umum didefinisikan sebagai sekumpulan teknik yang dapat diimplementasikan oleh *node* IPv6 untuk dapat kompatibel dengan *node* IPv4 yang sudah eksis sebelumnya. Berikut adalah beberapa mekanisme yang dikembangkan untuk Interkoneksi dari IPv4 ke IPv6.

DOI 10.21063/JTIF.2017.V5.2.63-72

2.7.1 Dual Stack

Dalam sebuah jaringan dual stack, host dan router mengimplementasikan IPv4 dan IPv6. Bagaimana tumpukan jaringan ganda dapat mendukung kedua layanan IPv4 dan IPv6 selama periode transisi. Jika komunikasi menggunakan IPv4 maka *interfac*e akan bertindak *interface* IPv4 murni, dan jika komunikasi menggunakan IPv6 maka *interface* akan bertindak sebagai *interface* IPv6 murni dan ketika kedua protokol ini digunakan secara bersama maka IPv4 dan IPv6 dapat berjalan secara bersamaan dengan perbedaan struktur alamat berbeda.

Dual Stack ini dalam pengkonfigurasi kedua prokotol tersebut harus secara terpisah dilakukanya seperti konfigurasi alamat statis maupun dinamis (DHCP untuk IPv4 atau DHCPv6 untuk IPv6),Routing dan komunikasi antar *client* (Ping untuk IPv4 atau Ping -6 untuk IPv6). Metode Transisi *dual stack* ini dalam protokol TCP/IP terletak pada layer internet yang bertangung jawab terhadap pada terkhusus pada Internet Protokol.

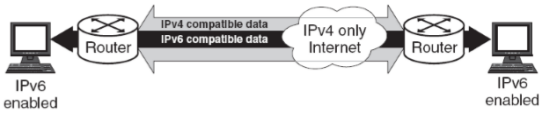


Gambar 4 TCP/IP *Dual Stack*

2.7.2 *Tunneling*

Tunneling digunakan untuk membangun jaringan IPv6 dengan memanfaatkan infrastruktur jaringan IPv4 yang sudah ada.Tunneling sering juga disebut enkapsulasi.

Dengan metode ini protokol IPv6 akan dienkapsulasi pada protokol IPv4. Paket yang terenkapsulasi ini kemudian di teruskan melalui jaringan IPv4 melalui infrastruktur jaringan IPv4.



Gambar 5 *Tunneling*

Pengimplentasian di Tunneling secara umum terbagi dua yaitu sebagai berikut :

a. *Manual Tunneling*

Manual Tunneling merupakan teknik tunneling yang mana interface tunnel dapat dikonfigurasi ekspelisit baik oleh administrator jaringan nya maupun memakai *Tunnel broker*. *Manual Tunneling* mudah untuk diimpementasikan namun pada suatu jaringan namun keterbatasan pada sisi keamanannya. Selain itu teknik sangat ketergantungan pada administrator untuk mengkonfigurasi ketika ada perubahan topologi jaringan.

b. *Automatic Tunneling*

Pada *Automatic Tunneling* interface *tunnel* memperoleh alamat atau *prefix* IPv6 berdasarkan fotmat alamat IPv4 yang telah sesuai dengan konfigurasi networknya, artinya prefix alamat IPv6 merupakan identitas unik yang dapat diperoleh dengan cara penintergrasian alamat IPv4 dengan mengubah format bilangan biner ke bilangan hexadecimal. Berikut ini salah satu *Automatic Tunneling* yaitu 6to4merupakan salah satu *tunneling* yang membolehkan IPv6 yang boleh melewati IPv4 dengan melakukan enskapsulasi dan deskapsulasi paket. Jenis tunneling ini dapat digunakan pada individual host maupun *local* IPv6 *network*. Pada automatic 6to4, interface tunnel harus memiliki alamat IPv6, prefix di 6to4 telah dialokasikan khusus 2002::/16 untuk keperluan tunneling dan tidak pernah digunakan untuk global unicast address

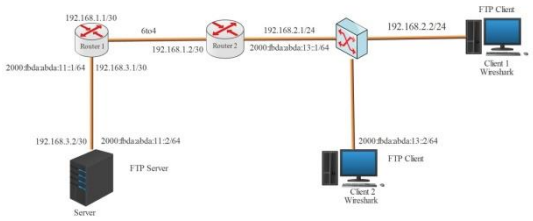
3. Pembahasan

3.1 Implementasi dan Pengujian

Implementasi topologi *Dual Stack* yang akan dianalisa, akan simulasikan seperti gambar dibawah ini. Dimana terdapat dua buah *router*, pertama router yang terhubung ke *server*nya dan router lainnya ke komputer *client* yang memiliki 2 komputer *Client*. Setiap *Interface* akan memiliki dua internet protokol (IP) yaitu

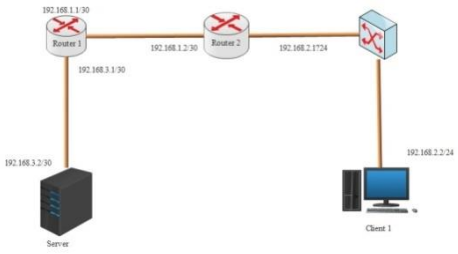
DOI 10.21063/JTIF.2017.V5.2.63-72

IPv4 dan IPv6 yang saling terkoneksi meskipun mempunyai struktur *header* yang berbeda. Berikut ini topologi implementasi Dual Stack IPv4 dan IPv6 di Mikrotik Router Os.



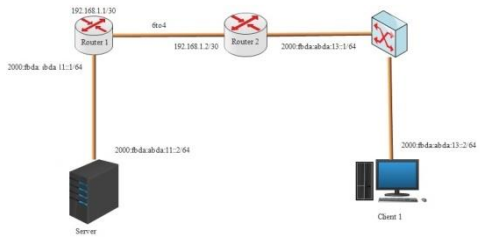
Gambar 6Topologi Pengujian *Dual Stack*

Untuk pengujian pada IPv4 hanya mengunakan satu client berikut gambar topologi pengujian IPv4 adalah :



Gambar 7 Topologi Pengujian IPv4

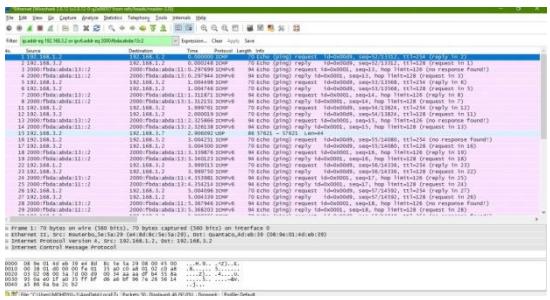
Dan berikut ini topologi pengujian IPv6 yang hanya mengunakan satu client komputer saja berikut adalah gambar Topologinya



Gambar 8 Topologi Pengujian IPv4

3.2 *Capture Dual Stack*

Hasil Captute merupakan analisa **Dual Stack** di sisi Aplikasi Wireshark yang mana akan terlihat bagaimana tumpukan 2 buah IP dalam satu interface yang sama dan pengujian kali hanya menggunakan Ping ke server ke dari masing-masing client dapat dilihat pada gambar 3.4



Gambar9*Capture Dual Stack*

3.3 Analisa Quality of Services (QoS)

Pada tahapan ini adalah menganalisa *Quality of Services* (Qos) pada jaringan Dual Stack di Mikrotik RouterOs. *Softwere* yang akan digunakan untuk menganalisa Qos adalah *Wireshark* 2.0.12, diasumsikan bahwa *SoftwereWireshark* 2.0.12 telah terpasang *client* yang akan digunakan pada saat proses analisa Qos-nya. Proses pengujiannya men-*download* file yang telah disediakan hanya menggunakan Protokol Internet satu saja yaitu IPv4 saja dan IPv6 saja dan menggunakan Dual Stack, pada setiap pengujian dilakukan sebanyak 10 kali untuk perukuran file uji yaitu 10 Mb, 20 Mb dan 50 Mb.

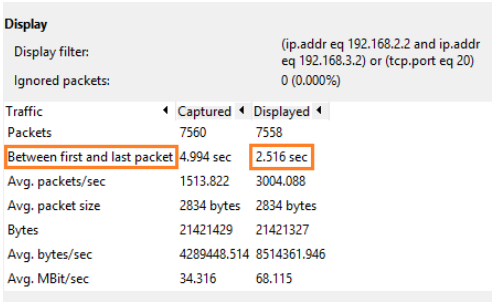
Pada tahapan ini adalah menganalisa *Quality of Services* (Qos) pada jaringan Dual Stack di Mikrotik RouterOs. *Softwere* yang akan digunakan untuk menganalisa Qos adalah *Wireshark* 2.0.12, diasumsikan bahwa *SoftwereWireshark* 2.0.12 telah terpasang *client* yang akan digunakan pada saat proses analisa Qos-nya. Proses pengujiannya men-*download* file yang telah disediakan hanya menggunakan Protokol Internet satu saja yaitu IPv4 saja dan IPv6 saja dan menggunakan *Dual Stack*, pada setiap pengujian dilakukan sebanyak 10 kali untuk perukuran file uji yaitu 10 Mb, 20 Mb dan 50 Mb.

3.3.1 Analisa *Transfer Time*

Transfer time adalah jumlah waktu yang dibutuhkan untuk mengirimkan seluruh paket dari *server* ke *client* yang dinyatakan dalam satuan *second*. Pengambilan parameter *transfer time* dilakukan dengan cara *download* file dari *server* ke *client*, pada saat itu juga *client* melakukan *capture data* di Wirehark melalui interface *Ethernet*, dan menghidari masuknya paket-paket selain paket yang berasal dari FTP oleh karena itu

DOI 10.21063/JTIF.2017.V5.2.63-72

akan *filter*ing pada saat *capture data* tersebut. Pengambilan nilai *transfer time* pada *summaryWireshark* seperti pada gambar 3.6 dibawah ini.



Gambar 10*Summary Transfer Time*

Pada gambar diatas data yang diambil adalah data yang hanya pada *displayed* bukan pada data *captured*. Data *transfer time* ini ambil dari perbedaan rentang waktu paket pertama sampai paket terakhir. Dari pengujian sebanyak 10 kali pada masing-masing file uji dan pengujian sebelum *Dual Stack* didapat dari nilai *transfer time* seperti pada tabel 3.1dibawah ini.

Tabel 1 Tabel Pengujian *Transfer Time* sebelum *Dual Stack*

Transfer Time (sec) Sebelum *Dual Stack*						
	IPv4			IPv6		
	10 MB	20 MB	50 MB	10 MB	20 MB	50 MB
1	1.348	2.882	7.877	1.586	2.988	6.101
2	1.398	3.044	6.016	1.451	3.193	6.002
3	1.376	2.688	5.678	1.753	2.929	5.782
4	1.353	2.442	5.411	1.414	2.304	6.203
5	1.513	2.483	6.108	1.443	3.052	6.248
6	1.456	2.385	5.761	1.517	2.518	6.366
7	1.492	2.371	5.483	1.602	2.606	5.824
8	1.482	2.290	5.287	1.385	3.203	5.930
9	1.393	3.092	5.272	1.554	3.098	5.501
10	1.402	3.192	5.877	1.584	3.008	5.992

Pada tabel 1 ini merupakan keseluruhan hasil pengujian tanpa menggunakan *Dual Stack* dari 10 kali pengujian di parameter *transfer time*. Untuk pengujian pada *Dual Stack* pada parameter *transfer time* diaktifkan tunnel 6to4 lalu akan dilalukan 10 kali pengujian download file dari *server* Xlight FTP *server* dapat dilihat pada tabel 3.2 dibawah ini.

Tabel 2 Tabel Pengujian *Transfer TimeDual Stack*

Transfer Time (sec) *Dual Stack*						
	IPv4			IPv6		
	10	20	50	10	20	50
1	1.663	3.232	7.540	1.715	2.873	5.886
2	1.477	2.697	5.7340	1.625	3.137	5.822
3	1.529	2.793	6.046	1.600	3.210	6.110
4	1.543	2.648	5.943	1.517	2.693	6.233
5	1.398	2.725	5.711	2.107	2.567	5.866
6	1.375	2.862	5.679	2.348	2.687	5.906
7	1.728	2.444	5.974	1.534	3.117	6.042
8	1.648	2.698	5.255	1.423	3.241	5.896
9	1.436	2.475	5.918	1.885	2.634	6.132
10	1.681	3.195	5.870	1.504	3.031	6.277

Pada Tabel 2 ini merupakan keseluruhan menggunakan *Dual Stack* dari hasil pengujian 10 kali *download file* dengan parameter *transfer time* dan berikut hasil rata-rata *transfer time* dari pengujian dengan sebelum *Dual Stack* dan menggunakan *Dual Stack* dapat dilihat pada tabel 3.3 dibawah ini.

Tabel 3 Rata-rata *Transfer Time* Sebelum *Dual Stack* dan Sesudah *Dual Stack*

Pengujian	*Transfer time (sec)*			
	Sebelum *Dual Stack*		*Dual Stack*	
	IPv4	IPv6	IPv4	IPv6
10576218 byte	1.421	1.528	1.547	1.728
20983103 byte	2.680	2.890	2.777	2.919
51948659 byte	5.855	5.995	5.987	6.017

Pada tabel 3 diatas dapat dilihat bahwa pengaruh kapasitas data terhadap *transfer time* maka semakin besar ukuran file maka semakin besar juga waktu yang dibutuhkan file untuk menhantarkan paket data. Pada keseluruhan pengujian file 10 Mb ,20 Mb dan 50 Mb, IPv4 memiliki *transfer time* yang lebih baik dibandingkan IPv6 yang memiliki nilai rata-rata *transfer time* sebesar 3.374 *sec* sedangkan IPv6 memiliki nilai rata-rata transfer time sebasar 3.513 *sec*, hal ini memunjukan bahwa konfigurasi jaringan dengan mengunakan IPv4 masih lebih baik dari pada IPv6. kali ini dikarena perbedaan struktur *header* antara IPv4 dengan IPv6, dimana panjang struktur IPv6 lebih panjang dua kali lipat dari *header* IPv4. Panjang *header* IPv6 sebesar 40 byte sedangkan IPv4 mempunyai *header* hanya 20 byte.

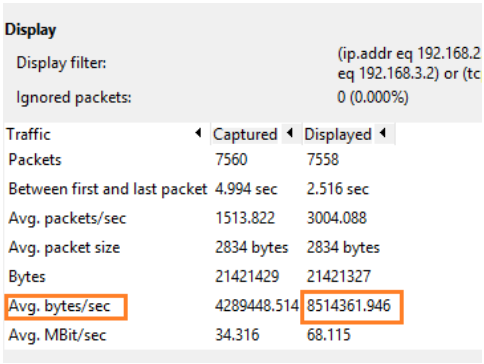
Dari pengujian pada jaringan *Dual Stack* lebih lambat *transfer time* dari pada tanpa menggunakan *Dual Stack*, yang memiliki nilai rata-rata *transfer time* sebesar 3.427 *sec* sedangkan tanpa menggunakan jaringan *Dual Stack* memiliki nilai rata-rata *transfer time* sebesar 3.391 *sec*, dikarenakan

DOI 10.21063/JTIF.2017.V5.2.63-72

banyaknya masuk paket data waktu peng-*capture data* dan ini perbedaannya tidak terlalu besar pengaruhnya dalam *transfer time*. Dari penjelasan diatas metode migrasi *Dual Stack* mudah diimplementasikan dan tidak teralu besar pengaruh pengunaan metode *Dual Stack* ini, dibandingkan hanya mengunakan satu *Internet Protokol* saja.

3.3.2 Analisa *Throughput*

Throughput merupakan nilai besaran kecepatan rata-rata dari sebuah file yang berhasil dikirimkan dari komputer *server* ke komputer *client* perdetiknya biasanya mengunakan satuan *bit per second*. Pengambilan parameter *throughput* dilakukan dengan cara *download* file dari *server* ke *client*, pada saat itu juga *client* melakukan *capture data* di Wirehark melalui interface Ethernet, dan menghidari masuknya paket-paket selain paket yang berasal dari FTP oleh karena itu akan *filter*ing pada saat *capture data* tersebut. Pengambilan nilai *transfer time* pada summary *Wireshark* seperti pada gambar 11 dibawah ini.



Gambar 11*Summary Throughput*

Pada gambar di atas dapat dilihat data yang akan di ambil adalah yang ada pada displayed bukan pada captured. Data-data yang tertangkap di *Wireshark* adalah dalam satuan bytes per *second* maka perlu diubah konversikan kesatuan *Megabyte per second* . Data keseluruhan dari 10 pengujian per masing-masing ukuran sebelum *Dual Stack* dengan parameter *throughput* yang dapat dilihat di tabel 3.4 dibawah ini.

Tabel 4 Tabel Pengujian*Throughput* sebelum *Dual Stack*

Throughput **(MBps) Sebelum *Dual Stack***						
	IPv4			**IPv6**		
	10	20	50	10	20	50
1	8.024	9.307	6.725	6.848	7.213	8.733
2	7.727	7.047	8.818	7.501	6.758	8.878
3	7.859	7.975	9.336	6.195	7.373	9.212
4	7.997	8.777	9.788	7.680	7.076	8.591
5	7.148	8.640	8.578	7.548	8.570	8.531
6	7.425	8.999	9.349	7.162	9.364	8.358
7	7.253	9.046	9.901	6.778	6.737	9.142
8	7.299	9.370	10.174	7.842	8.269	8.985
9	7.766	7.072	10.014	7.001	6.968	9.613
10	7.515	6.722	9.026	6.872	7.177	8.894

Pada tabel 4 ini merupakan keseluruhan hasil pengujian tanpa menggunakan *Dual Stack* dari 10 kali pengujian di parameter *throughput*. Untuk pengujian pada *Dual Stack* pada parameter *throughput* diaktifkan tunnel 6to4 lalu akan dilalukan 10 kali pengujian *download* file dari *server* Xlight FTP *server* dapat dilihat pada tabel 5 dibawah ini.

Tabel 5 Tabel Pengujian*ThroughputDual Stack*

Throughput **(MBps) *Dual Stack***						
	IPv4			IPv6		
	10	20	50	10	20	50
1	6.493	6.640	9.598	6.349	7.519	9.064
2	7.313	7.949	7.032	6.878	6.878	9.038
3	7.057	7.681	9.255	6.809	8.028	8.744
4	7.012	8.098	8.931	7.162	6.726	9.174
5	7.727	7.876	8.774	5.169	8.411	8.508
6	7.867	7.494	9.284	4.642	8.035	9.108
7	6.259	8.776	9.340	7.100	6.662	8.576
8	6.565	7.954	8.881	7.657	6.930	8.703
9	7.530	6.717	8.965	5.778	8.824	8.824
10	6.433	8.672	9.043	7.242	9.044	9.044

Pada Tabel 5 ini merupakan keselurahan menggunakan *Dual Stack* dari hasil pengujian 10 kali mendownload file dengan parameter *throughput* dan berikut hasil rata-rata *transfer time* dari pengujian dengan sebelum *Dual Stack* dan menggunakan *Dual Stack* dapat dilihat pada tabel 6 dibawah ini.

Tabel 3.6 Rata-rata *Transfer Time* Sebelum *Dual Stack* dan Sesudah *Dual Stack*

File Pengujian	***Throughput*** **(MBps)**			
	Sebelum *Dual Stack*		***Dual Stack***	
	IPv4	**IPv6**	**IPv4**	**IPv6**
10576218 byte	7.621	7.142	7.026	6.450
20983103 byte	8.296	7.550	7.786	7.450
51948659 byte	9.159	8.902	8.910	8.878

Pada tabel di atas dapat dilihat bahwa pengaruh kapasitas data terhadap *throughput* maka semakin besar ukuran file maka semakin besar juga *throughput* yang dihantarkan dari *server* ke komputer *client*.

DOI 10.21063/JTIF.2017.V5.2.63-72

Pada keseluruhan pengujian file 10 Mb ,20 Mb dan 50 Mb, IPv4 lebih baik dibandingkan mengunakan IPv6, yang memiliki nilai rata-rata *throughput* sebesar 8.317 MBpssedangkan IPv6 memiliki nilai rata-rata *throughput* sebesar 7.729 MBps, hal ini menunjukan bahwa konfigurasi jaringan dengan mengunakan IPv4 masih lebih baik dari pada parameter *throughput* ini dikarena perbedaan struktur *header* antara IPv4 dengan IPv6, dimana panjang struktur IPv6 lebih panjang dua kali lipat dari *header* IPv4. Panjang *header* IPv6 sebesar 40 byte sedangkan IPv4 mempunyai *header* hanya 20 byte.

Dari pengujian pada jaringan *Dual Stack* lebih lambat *throughput* dari pada tanpa *Dual Stack*, yang memiliki nilai rata-rata *throughput* sebesar 7.750 MBps sedangkan tanpa menggunakan jaringan *Dual Stack* memiliki nilai rata-rata *throughput* sebesar 8.296 MBps, Ini dikarenakan banyaknya masuk paket data waktu peng-*capture data* dan ini perbedaannya tidak terlalu besar pengaruhnya dalam *throughput*. Dari penjelasan diatas metode migrasi *Dual Stack* mudah diimplementasikan dan tidak teralu besar pengaruh pengunaan metode *Dual Stacks* ini dibandingkan hanya mengunakan satu *Internet Protokol* saja.

3.3.3 Analisa *Delay*

Delay merupakan waktu tunda yang diperlukan oleh paket data dari awal paket dikirimkan sampai hingga paket sampai ketujuan. Parameter delay dihitungan dengan cara membagi transfer time dengan jumlah bit data dengan rumus dibawah ini.

$$Delay = \frac{Transfer\ Time}{Jumlah\ bit}$$

Hasil pengambilan data secara keseluruhan untuk rata-rata parameter delay dari hasil perhitungan *transfer time* dibagi dengan jumlah bit yang terbaca melalui aplikasi *Wireshark* dan semakin kecil nilai dari delay baik performa jaringan. Data rata-rata delay dadapat dilihat pada tabel 4.7 dibawah ini.

Tabel 6 Rata-rata *Transfer Time* Sebelum *Dual Stack* dan Sesudah *Dual Stack*

File Pengujian	*Delay (usecond)*			
	Sebelum *Dual Stack*		*Dual Stack*	
	IPv4	IPv6	IPv4	IPv6
10576218 byte	0.0167	0.0181	0.0182	0.0204
20983103 byte	0.0159	0.0172	0.0165	0.0174
51948659 byte	0.0141	0.0144	0.0143	0.0144

Pada tabel di atas dapat dilihat bahwa pengaruh kapasitas data terhadap *delay* tidak mempengaruhi nilai dari juga juga delay tersebut . Pada keseluruhan pengujian file 10 Mb ,20 Mb dan 50 Mb, IPv4 memiliki *delay* yang lebih kecil dari IPv6, yang memiliki nilai rata-rata *delay* sebesar 0.0160 µ*sec* sedangkan IPv6 memiliki nilai rata-rata delay sebesar 1.070 µ*sec*, hal ini memunjukan bahwa konfigurasi jaringan dengan mengunakan IPv4 masih lebih baik dari pada parameter *delay* ini dikarena perbedaan struktur *header* antara IPv4 dengan IPv6, dimana panjang struktur IPv6 lebih panjang dua kali lipat dari *header* IPv4. Panjang *header* IPv6 sebesar 40 byte sedangkan IPv4 mempunyai *header* hanya 20 byte.

Dari pengujian pada jaringan *Dual Stack* memiliki lebih lambat *delay* dari pada tanpa *Dual Stack*, yang memiliki nilai rata-rata *delay* sebesar 0.01609 µ*sec* sedangkan tanpa *Dual Stack* memiliki nilai rata-rata *throughput* sebesar 0.01692 µ*sec*, dikarenakan banyaknya masuk paket data waktu peng-*capture data* dan ini perbedaannya tidak terlalu besar pengaruhnya dalam *delay* Dari penjelasan diatas metode migrasi *Dual Stack* mudah diimplementasikan dan tidak teralu besar pengaruh pengunaan metode *Dual Stack* ini dibandingkan hanya megunakan satu *Internet Protokol* saja,

4. Penutup

4.1 Kesimpulan

Dari pengujian yang dilakukan, untuk menginterkoneksikan IPv4 dengan IPv6 menggunakan Tunnel 6to4 dikarenakan Dual Stack tidak bisa terkoneksi secara langsung antara IPv4 dengan IPv6. Dan proses migrasi antara IPv4 ke IPv6 tidak mengalami *problem* disisi implementasi. Dari hasil analisa *Quality of Services* (Qos) pada parameter *transter time*, IPv4 memiliki *transfer time* yang lebih baik dibandingkan dengan menggunakan IPv6, yang memiliki nilai rata-rata *transfer time* 3.374 *sec*.

DOI 10.21063/JTIF.2017.V5.2.63-72

Sedangkan IPv6 memiliki nilai rata-rata transfer time sebasar 3.513 *sec* dan pengujian *Dual Stack* lebih lambat *transfer time* dari tanpa menggunakan *Dual Stack,* yang memiliki nilai rata-rata 3,989 sec sedangkan tanpa *Dual Stack* memiliki nilai rata-rata *transfer time* 3.488 *sec*. Pada pengujian Parameter *throughput* semakin besar kapasitas sebuah file maka akan semakin besar nilai *throughput* yang dihantarkan dari *server* ke *client*, pada IPv4 lebih baik dibandingkan menggunakan IPv6, yang memiliki nilai rata-rata *throughput* 8.330 MBps.Sedangkan IPv6 memiliki nilai rata-rata *throughput* 7.729 MBps dan penggunaan *Dual Stack* lebih lambat *throughput* dari pada tanpa menggunakan *Dual Stack,* yang memiliki nilai rata-rata *throughput* 7.769 MBps. Sedangkan tanpa *Dual Stack* memiliki nilai rata-rata *throughput* 8.289 MBps. Pada pengujian parameter *delay* perbedaan besarnya kapasitas sebuah file tidak menpengaruhi nilai dari *delay*, IPv4 memiliki *delay* yang lebih kecil dari IPv6, yang memiliki nilai rata-rata *delay* 0.0160 µ*sec.* Sedangkan pada IPv6 memiliki nilai rata-rata *delay* 1.070 µ*sec* dan penggunaan *Dual Stack* memiliki lebih lambat *delay* dari pada tanpa menggunakan *Dual Stack*, yang memiliki nilai rata-rata *delay* 0.01687 µ*sec.* Sedangkan tanpa *Dual Stack* memiliki nilai rata-rata *throughput* 0.01613 µ*sec*. Dari analisa *Quality of Servies* (QoS) penggunaan metode *Dual Stack* tidak terlalu ada pangaruh yang signifikan terhadap performa jaringan.

4.2 Saran

Pada penelitian mengunakan metode *Dual Stack*ini hanya menguji *Quality of Services (QoS)* pada jaringan *private* dan untuk routing protokol menggunakan *routing static*. Untuk penelitian lebih lanjut maka disarankan untuk mencoba mengunakan jaringan *public* dan mengunakan *dynamic routing* seperti OSPF v3 yang bisa menggunakan IPv6. Dan dapat menambah jumlah *router* Mikrotik Os lebih banyak.

Daftar Pustaka

Artondo, Reko. *Analisa dan Implementasi IPv6 Tunnel Broker Untuk Interkoneksi IPv6 dan IPv4*.Semarang : Universitas Diponegoro

Gin, Gin Yugianto, Oscar Rachman. 2012. *Router Teknologi, Konsep,KonfigurSasi, dan Troubleshoting Berbasis Windows, Cisco, Linux, MacOS dan Mikrotik Router*. Bandung : Informatika

Pratama,I Putu Agus Eka. 2014. *Handbook Jaringan Komputer Teori dan Praktik Berdasarkan Open Source*. Bandung : Informatika.

Purbo, Onno W , Raharjo dan Sarujih. 2016. *IPv6 Fondasi Internet Masa Depan*. Andi Publisher.

Rachman, Oscar, Gin Gin Yugianto. 2008. *Tcp/Ip Dalam Dunia Informatika dan Telekomunnikasi.* Bandung : Informatika

Ramadhan, Gilang. 2010. *Analisa PerbandinganPerformasi Jaringan IPv4, IPv6 dan Tunnel 6to4 untuk Aplikasi File Tranfer File Protokol (FTP) pada Media Wired dan Wireless pada sisi Client*. Depok : Universitas Indonesia

Towidjojo, Rendra. 2012. *Konsep Dan Impementasi Routing Dengan Router Mikrotik*. Jasakom

Wardoyo, Siswo. 2014. *Analisa Performa File Transport Protokol pada perbandingan Metode IPv4 Murni, IPv6 Murni dan TUNNELING 6to4 Berbasis Router Mikrotik*. Cilegon : Universitas Sultan Agung Tirtayasa.

https://id.wikipedia.org/wiki/IPv6 (diakses tanggal 15 April 2017)

https://id.wikipedia.org/wiki/Alamat_IP_versi_6 (diakses tanggal 15 April 2017)

https://docs.oracle.com/cd/E18752_01/html/816-4554/ipv6-overview-10.html(diakses tanggal 28 April 2017)

DOI 10.21063/JTIF.2017.V5.2.63-72