BAB 2

DASAR TEORI

2.1. KAJIAN PUSTAKA

Penelitian yang dikaji oleh Okky Dwiana Wibowo, Fazmah Arif Yulianto, Gandeva Bayu Satrya tentang "Analisis QOS Dengan Metode Differentiated Services (DIFFSERV) Dalam Sistem IPv6-Over-IPv4 Gre Tunnel meneliti tentang teknik transisi menuju IPv6 tanpa mengganggu infrastruktur IPv4 yang sudah ada dengan tunneling IPv6-over-IPv4 dengan menggunakan metode Generic Routing Encapsulation (GRE) yang dikembangkan oleh *Cisco System* [3].

Penelitian pada tahun 2015, oleh Priya Bali yang berjudul "*A Detail Comprehensive Review on IPv4-to-IPv6 Transition and Co-Existence Strategies*". Di dalam penelitian ini, menunjukkan perpindahan IPv4 dan IPv6 menggunakan metode *dual stack* dan *tunel* sebagai penyambung antar jaringan tersebut [4]. Hasil yang didapatkan dalam penelitian tersebut menjelaskan, menggunakan NAT64 dalam perpindahan IPv4 ke IPv6 menjadi jalan terbaik meski untuk sementara waktu sampai perpindahan IPv6 telah menyeluruh.

Penelitian pada tahun 2016, oleh Audy Septarindra, Rendy Munadi, Ridha Muldina Negara yang berjudul "Implementasi Dan Analisis Performa *Multi-Protocol Label Switching-Virtual Private Network* (MPLS-VPN) Dengan *Metode Generic Routing Encapsulation* Pada Layanan Berbasis File Transfer Protocol (FTP)" meneliti tentang performansi MPLS melalui GRE untuk enkapsulasi paket MPLS dalam terowongan IP. Enkapsulasi MPLS dalam IP *tunnels* membuat *link virtual point-to-point* di seluruh jaringan *non*-MPLS [5].

Penelitian pada tahun 2018, oleh Muhammad Ismu Haji, Sugeng Purwantoro E. S. G. S, dan Satria Perdana Arifin yang berjudul "*Analysis Tunneling IPv4 and IPv6 on VoIP Network*" meneliti tentang perbandingan perfomansi VoIP pada jaringan IPv6 dan IPv4 [6]. Ditemukan bahwa nilai rata-rata untuk IPv4 dan IPv6 tidak menunjukan hasil yang signifikan, namun nilai QoS pada IPv6 menunjukan kategori lebih baik dibandingkan dengan IPv4.

Pada penelitian yang akan dilakukan oleh penulis, penulis akan mengimplementasikan penggunaan GRE IPv6 Tunnel menggunakan *routing static*

. Kinerja yang akan di analisis adalah QoS pada layanan *VoIP* dengan menggunakan aplikasi pihak ketiga dan mengukur parameter *throughput, delay, jitter* dan *packet loss.*

2.2. DASAR TEORI

2.2.1 *Generic Routing Encapsulation* IPv6 (GRE 6)

Fitur GRE IPv6 Tunnels memungkinkan pengiriman paket dari protokol lain melalui jaringan IPv6 dan memungkinkan perutean paket IPv6 antara jaringan pribadi di seluruh jaringan publik dengan dialihkan secara global Alamat IPv6. Generic routing encapsulation (GRE) adalah protokol unicast yang menawarkan kelebihan merangkum siaran dan lalu lintas multicast (streaming multicast atau protokol perutean) atau non-IP lainnya protokol dan dilindungi oleh IPsec.

Untuk terowongan GRE point-to-point, setiap antarmuka terowongan membutuhkan sumber alamat IPv6 terowongan dan sebuah terowongan alamat IPv6 tujuan ketika sedang dikonfigurasi. Semua paket dienkapsulasi dengan *header* IPv6 luar dan *heade*r GRE.

Perlindungan terowongan GRE IPv6 memungkinkan perangkat berfungsi sebagai gateway keamanan, membuat terowongan IPsec di antaranya perangkat gateway keamanan lainnya, dan memberikan perlindungan cryptto IPsec untuk lalu lintas dari jaringan internal saat lalu lintas dikirim melalui Internet IPv6 publik. Fungsionalitas perlindungan terowongan GRE IPv6 mirip dengan model gateway keamanan yang menggunakan perlindungan terowongan IPv4 GRE [7].

2.2.2 *Internet Protocol version* (IPv6)

Jaringan akses yang dikembangkan dalam beberapa tahun ke depan sebagian besar akan hanya IPv6. Untuk memberikan akses *dual-stack* ke jaringan tersebut, kami membutuhkan mekanisme *tunneling IPv4-over-IPv6* yang dapat mempertahankan ketersediaan IPv4 ketika jaringan akses beralih ke IPv6, dan karenanya secara signifikan memajukan transisi [8].

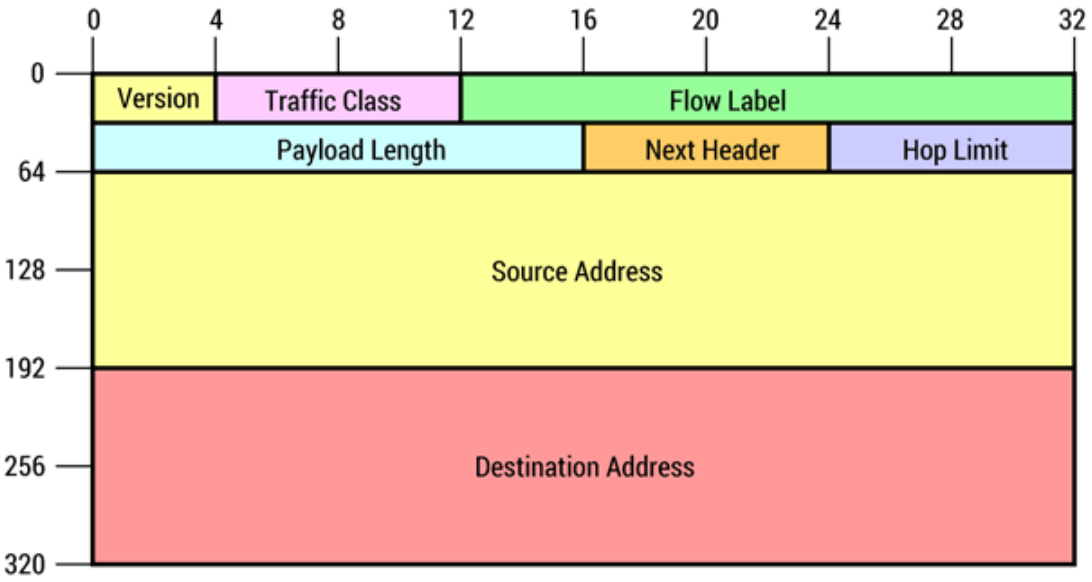
Penelitian *tunneling* awal berfokus pada pengembangan teknik *IPv6-over-IPv4*. Baru-baru ini, dengan pertumbuhan jaringan akses IPv6, komunitas Internet telah mengusulkan serangkaian teknik tunneling *IPv4-over-IPv6*, termasuk *Public 4over6*, *Dual-Stack Lite*, dan 4, untuk memenuhi berbagai tuntutan transisi. ISP dapat mengalokasikan alamat IPv4 untuk mengakhiri *host* melalui jaringan akses

IPv6. Hal ini memungkinkan ISP menjaga fleksibilitas operasi IPv6 dengan biaya AFBR mempertahankan pemetaan alamat per-host untuk enkapsulasi. Publik 4over6, yang menyediakan akses IPv4 ke pengguna melalui jaringan IPv6 dengan alamat IPv4 publik, sesuai dengan kasus ini [8].

Saat mengalokasikan alamat IPv4, konsentrator 4over6 mengelola pemetaan antara alamat IPv4 yang dialokasikan dan alamat IPv6 pelanggan. Prosedur penerusan data adalah prosedur terowongan IPv4-in-IPv6 standar, dan pemetaan digunakan untuk pencarian alamat tujuan selama proses enkapsulasi pada konsentrator. Pemetaan yang dikelola dalam 4over6 *concentrator* adalah pemetaan alamat per pelanggan, sehingga layanan IPv4 yang disediakan *Public* 4over6 sepenuhnya dua arah [8].

a) Format *Header* IPv6

Pada IPv6 digunakan *header* paket yang sederhana, dengan *header* yang sederhana ini paket dapat di proses lebih cepat. *Header* pada IPv6 merupakan penyederhanaan dari *header* IPv4 dengan cara menghilangkan bagian yang tidak diperlukan atau jarang digunakan dan menambahkan bagian yang lebih baik [9].



Gambar 2. 1 Format *Header* pada IPv6 [9].

b) *Prefix* pada IPv6

Pengalamatan IPv4 saat dalam notasi *dotted-decimal* format dapat direpresentasikan dengan menggunakan angka *prefix* pada *subnet mask*. IPv6 juga mempunyai angka *prefix*, tetapi tidak digunakan untuk *subnet mask*, karena

memang IPv6 tidak punya *subnet mask, subnet mask* prefiks yaitu sebuah bagian dari pengalamatan IP, dimana bit yang memiliki nilai yang tetap atau bit tersebut bagian dari subnet identifier. *Prefix* dalam IPv6 sama seperti halnya *prefix* alamat IPv4, yaitu [alamat] atau [angka panjang *prefix*]. Panjang *prefix* ini menentukan jumlah bit terbesar dari paling kiri yang membuat prefik *subnet*. Contohnya sebuah alamat IPv6 prefix dapat dituliskan sebagai berikut : 19:19::/64 Pada contoh ini 64 bit pertama dari alamat tersebut sebagai prefiks alamat, sementara 64 bit sisanya akan dianggap sebagai interface ID [9].

c) Struktur paket data IPv6

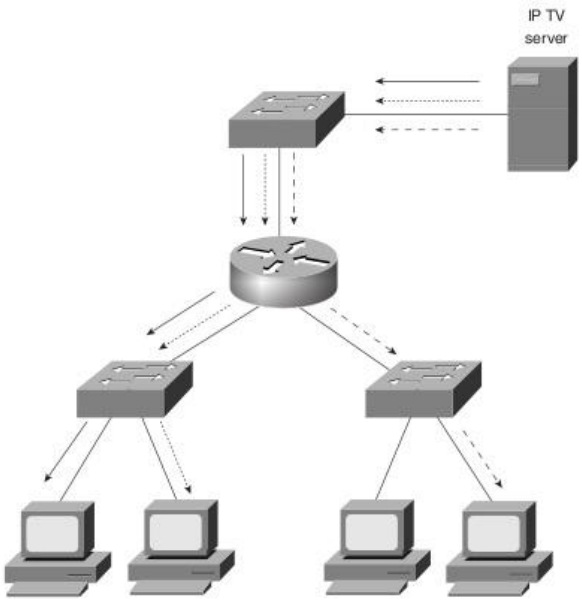
Dalam perancangan *header packet* ini, diupayakan agar *cost*/nilai pemrosesan *header* menjadi kecil untuk mendukung komunikasi data yang lebih *real time*. Misalnya, *address* awal dan akhir menjadi dibutuhkan pada setiap *packet.* Sedangkan pada *header* IPv4 ketika *packet* dipecah-pecah, ada *field* untuk menyimpan urutan antar *packet*. Namun *field* tersebut tidak terpakai ketika *packet* tidak dipecah-pecah. *Header* pada Ipv6 terdiri dari dua jenis, yang pertama, yaitu *field* yang dibutuhkan oleh setiap packet disebut *header* dasar, sedangkan yang kedua yaitu *field* yang tidak selalu diperlukan pada *packet* disebut *header* ekstensi, dan *header* ini didefinisikan terpisah dari header dasar. *Header* dasar selalu ada pada setiap *packet*, sedangkan *header* tambahan hanya jika diperlukan diselipkan antara header dasar dengan data. *Header* tambahan, saat ini didefinisikan selain bagi penggunaan ketika packet dipecah, juga didefinisikan bagi fungsi sekuriti dan lain-lain. *Header* tambahan ini, diletakkan setelah header dasar, jika dibutuhkan beberapa header maka header ini akan disambungkan berantai dimulai dari header dasar dan berakhir pada data. *Router* hanya perlu memproses header yang terkecil yang diperlukan saja, sehingga waktu pemrosesan menjadi lebih cepat. Hasil dari perbaikan ini, meskipun ukuran header dasar membesar dari 20 bytes menjadi 40 bytes namun jumlah *field* berkurang dari 12 menjadi 8 buah saja [10].

d) Alamat IPv6

Pada IPv6 pengalamatan dibedakan menjadi 3 yaitu, *Unicast Address* (*one to one*), *Multicast* (*one to many*), dan *Anycast Address*.

1) *Unicast Address* (*one to one*)

Digunakan untuk komunikasi satu lawan satu, dengan menunjuk satu *host*. Pada *Unicast address* ini terdiri dari 3 jenis. *Global Address* yang digunakan misalnya untuk *address provider* atau *address geografis*. *Link Local Address* adalah *address* yang dipakai di dalam satu *link* saja. Yang dimaksud *link* di sini adalah jaringan lokal yang saling tersambung pada satu *level*. *Address* ini dibuat secara otomatis oleh *host* yang belum mendapat *address global*, terdiri dari 10+n *bit prefix* yang dimulai dengan "FE80" dan *field* sepanjang 118-n bit yang menunjukkan nomor *host*. *Link Local Address* digunakan pada pemberian IP *address* secara otomatis. *Site-Local, Address* yang setara dengan *private address*, yang dipakai terbatas di dalam *site* saja. *Address* ini dapat diberikan bebas, asal unik di dalam *site* tersebut, namun tidak bisa mengirimkan *packet* dengan tujuan alamat ini di luar dari *site* tersebut.

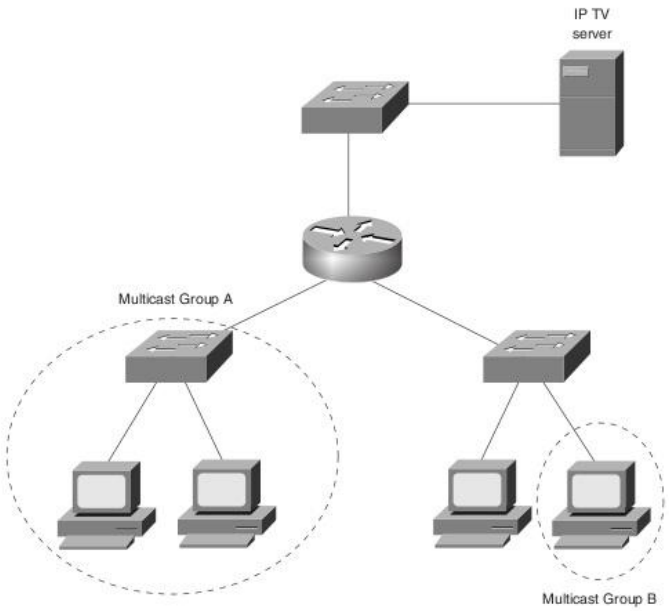


Gambar 2. 2 Pengiriman paket pada *Unicast Address* [10].

2) *Multicast* (*one to many*)

Digunakan untuk komunikasi 1 lawan banyak dengan menunjuk *host* dari *group*. *Multicast Address* ini pada IPv4 didefinisikan sebagai kelas D, sedangkan pada IPv6 ruang yang 8 bit pertama nya di mulai dengan "FF" disediakan untuk *multicast Address*. Ruang ini kemudian dibagi-bagi lagi untuk menentukan *range* berlakunya. Kemudian *Blockcast address* pada IPv4 yang address bagian host nya didefinisikan sebagai "1", pada IPv6 sudah termasuk di

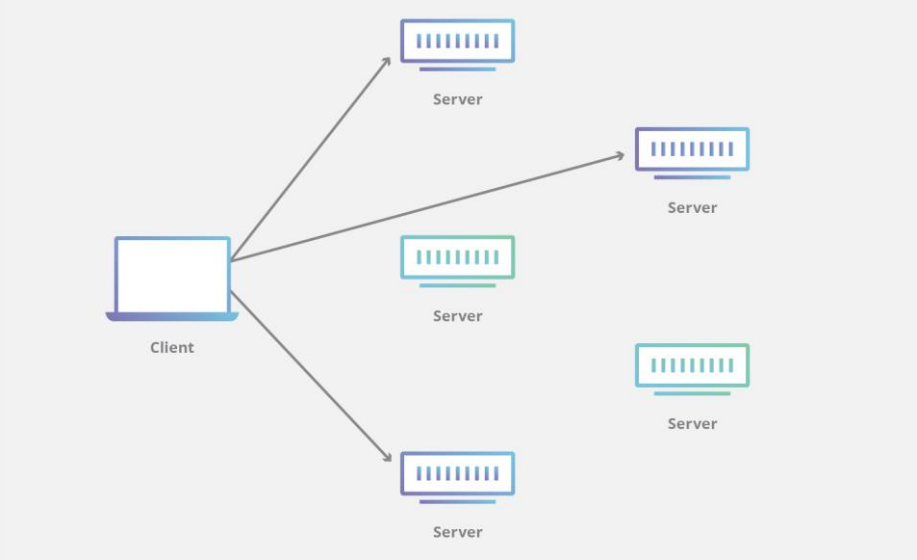
dalam *multicast Address* ini. *Blockcast address* untuk komunikasi dalam segmen yang sama yang dipisahkan oleh *gateway,* sama halnya dengan *multicast address* dipilah berdasarkan *range* tujuan [10].



Gambar 2. 3 Pengiriman paket pada Multicast Address [10].

3) *Anycast Address*

Menunjuk *host* dari *group*, tetapi *packet* yang dikirim hanya pada satu *host* saja. Pada *address* jenis ini, sebuah *address* diberikan pada beberapa *host*, untuk mendefinisikan kumpulan *node*. Jika ada packet yang dikirim ke *address* ini, maka *router* akan mengirim *packet* tersebut ke *host* terdekat yang memiliki *Anycast address* sama. Dengan kata lain pemilik packet menyerahkan pada router tujuan yang paling "cocok" bagi pengiriman packet tersebut. Pemakaian *Anycast Address* ini misalnya terhadap beberapa *server* yang memberikan layanan seperti DNS (*Domain Name Server*). Dengan memberikan *Anycast Address* yang sama pada server-server tersebut, jika ada packet yang dikirim oleh *client* ke *address* ini, maka router akan memilih server yang terdekat dan mengirimkan *packet* tersebut ke server tersebut. Sehingga, beban terhadap server dapat terdistribusi secara merata. Bagi *Anycast Address* ini tidak disediakan ruang khusus. Jika terhadap beberapa host diberikan sebuah address yang sama, maka address tersebut dianggap sebagai *Anycast Address* [10].



Gambar 2. 4 Pengiriman paket pada *Any cast Address*. [10].

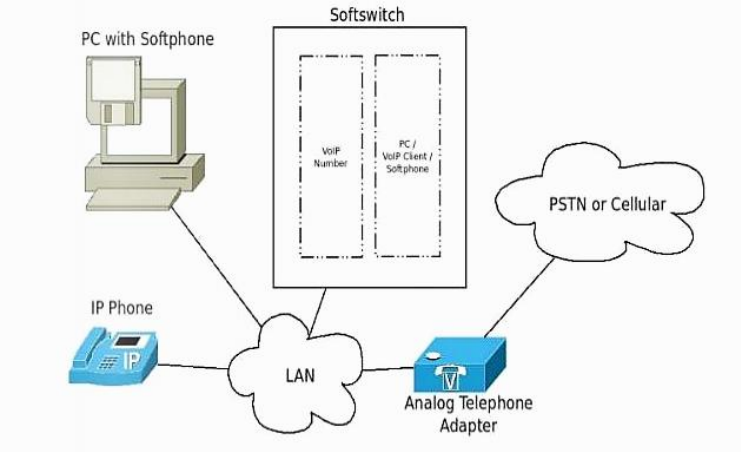
e) Keuntungan IPv6

Otomatisasi berbagai *setting*/*Stateless-less auto-configuration* (*plug and play*) *Address* pada IPv4 pada dasarnya statis terhadap *host*. Biasanya diberikan secara berurut pada *host*. Memang saat ini hal di atas bisa dilakukan secara otomatis dengan menggunakan DHCP (*Dynamic Host Configuration Protocol*), tetapi hal tersebut pada IPv4 merupakan fungsi tambahan saja, sebaliknya pada IPv6 fungsi untuk men setting secara otomatis disediakan secara standar dan merupakan default nya. Pada *setting* otomatis ini terdapat 2 cara tergantung dari penggunaan address, yaitu setting otomatis *stateless* dan *state full* [10].

2.2.3 *Voice Over Internet Protocol* (VoIP)

Teknologi VoIP menjadi dasar dari *Next Generation Network* (NGN) maupun jaringan selular 4G yang digunakan oleh operator telekomunikasi masa datang. Teknik VoIP di adopsi oleh rekan-rekan Amatir Radio (ORARI) untuk menggunakan internet sebagai *relay* jarak jauh. Teknik VoIP di Amatir Radio dikenal sebagai eQSO. Inti dari VoIP terdapat pada jantung VoIP yaitu jaringan *softswitch*, yang menyimpan semua informasi tentang pelanggan. Dalam pandangan sederhana, VoIP *softswitch* pada dasarnya memiliki tabel pemetaan nomor telepon pelanggan dan komputer atau IP alamat pelanggan. Jika ada salah satu pelanggan yang ingin melakukan panggilan maka pelanggan tersebut meminta pada *Softswith* untuk mengetahui alamat dan tujuan pelanggan yang lain, alamat

tujuan dapat menjadi alamat IP, pada dasarnya *softswitch* tempat berkumpulnya semua nomor telepon pelanggan dan IP alamat [11].



Gambar 2.5 Cara Kerja VoIP [11]**.**

Protokol-protokol yang menunjang terjadinya komunikasi VOIP adalah

1) *Transmission Control Protocol* (TCP)

TCP merupakan protokol yang *connectionoriented* yang artinya menjaga reliabilitas hubungan komunikasi *end-to-end*. Konsep dasar cara kerja TCP adalah mengirim dan menerima segmen-segmen informasi dengan panjang data bervariasi pada suatu datagram internet, TCP menjamin reliabilitas hubungan komunikasi karena melakukan perbaikan terhadap data yang rusak hilang atau kesalahan kirim [11].

2) *User Datagram Protocol* (UDP)

UDP merupakan salah satu protokol utama di atas IP dan merupakan transport protocol yang lebih sederhana dibandingkan dengan TCP. UDP digunakan untuk situasi yang tidak mementingkan mekanisme reliabilitas, artinya pada protokol UDP ini komunikasi akan tetap berlangsung tanpa memperdulikan koneksi antara sumber dan tujuan [11].

3) *Internet Protocol* (IP)

Internet Protocol adalah protokol lapisan jaringan (*network layer* dalam *OSI Reference Model*) atau protokol lapisan *internet work* (*internetwork layer* dalam *DARPA Reference Model*) yang digunakan oleh protokol TCP/IP untuk melakukan pengalamatan dan *routing* paket data antar *host* di jaringan komputer berbasis TCP/IP [11].

4) H.323

H.323 adalah salah satu dari rekomendasi ITU-T (*International Telecommunications Union Telecommunications*). H.323 merupakan standar yang menentukan komponen, protokol, dan prosedur yang menyediakan layanan komunikasi *multimedia*, layanan tersebut adalah komunikasi *audio, video,* dan data *real-time*, melalui jaringan berbasis paket (*packet based network*) [11].

5) *Session Initiation Protocol* (SIP)

SIP adalah suatu *signaling protocol* pada *layer* aplikasi yang berfungsi untuk membangun, memodifikasi, dan mengakhiri suatu sesi *multimedia* yang melibatkan satu atau beberapa pengguna 3. Sesi *multimedia* adalah pertukaran data antar pengguna yang bisa meliputi suara, *video*, dan *text*, IP tidak menyediakan layanan secara langsung, tetapi menyediakan pondasi yang dapat digunakan oleh protokol aplikasi lainnya untuk memberikan layanan yang lengkap bagi pengguna, misalnya dengan RTP (*Real Time Transport Protocol*) untuk transfer data secara *real-time* [11].

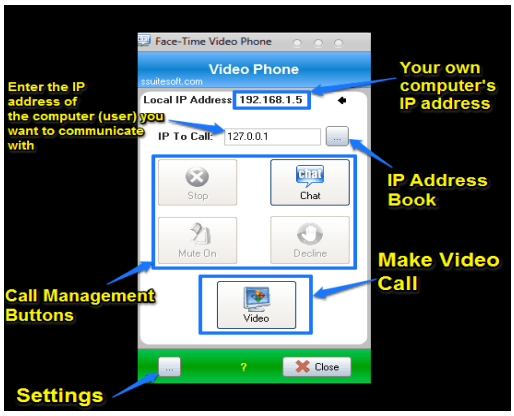
2.2.4 *Routing Static*

Routing merupakan proses penentuan jalur terbaik (*best path*) untuk mencapai suatu tujuan *network*. *Routing* juga dapat berarti proses memindahkan suatu paket data dari *host* pengirim ke *host* tujuan dimana *host* pengirim dan *host* tujuan tidak berada dalam satu *network*. Bila mengacu kepada pemodelan OSI (*Open System Interconnection*), maka proses *routing* terjadi pada *layer* 3 (*Network Layer*), oleh karena itu *routing* erat sekali kaitannya dengan IP *Address* atau pengalamatan IP [12].

Routing static adalah sebuah teknik *routing* yang dilakukan dengan memasukkan *entry route* ke *network* tujuan (*remote network*) ke dalam tabel *routing* secara manual oleh *administrator* jaringan. Bila sebuah *router* memiliki satu *remote network*, maka *administrator* jaringan harus memasukkan satu *entry route* ke *network* tersebut. Dalam memasukkan *entry route administrator* harus mengetahui dengan pasti *gateway* yang akan digunakan untuk mencapai *remote network*. Untuk jaringan yang terdiri dari beberapa *router*, maka penentuan *gateway* maupun jalur (*path*) harus dilakukan dengan baik [12].

2.2.5 *SSuite FaceTime*

SSuite Face Time Video Phone adalah aplikasi video LAN gratis yang sangat berguna yang memungkinkan komunikasi peer-to-peer dengan orang lain, menggunakan *Voice over Internet Protocol* (VoIP). *SSuite Face Time* ini bekerja pada jaringan LAN dan internet. Perangkat yang dibutuhkan hanya headset dengan mic, webcam, dan konektivitas jaringan. *SSuite Face Time* pada dasarnya berfungsi dengan membuat koneksi peer-to-peer dengan komputer pengguna lain di jaringan, menggunakan alamat IP komputer tersebut. *Software* ini dapat mencari semua komputer yang terhubung ke jaringan yang sama, dan *software* ini dapat menambahkan alamat IP komputer lain ke daftar kontak [13].



Gambar 2. 6 Tampilan *SSuite Face Time* [13].

2.2.6 *Quality of Service* (*QoS*)

Parameter QoS menggolongkan kualitas transfer yang diberikan oleh suatu koneksi yang diperoleh dengan membandingkan unit data pada sisi 5 masukan dan keluaran interface. Parameter QoS adalah: [14]

a) *Packet Loss*

Packet loss merupakan parameter yang menggambarkan kondisi yang menunjukkan jumlah paket yang hilang [15]. Besar *packet Loss* dapat diukur dengan cara sebagai berikut:

$$\textbf{\textit{Packet Loss}} = \frac{\text{Total paket yang dikirim} - \text{total paket yang diterima}}{\text{Total paket yang dikirim}} \quad (\mathbf{2.1})$$

b) *Delay*

Delay adalah permasalahan umum yang terjadi pada jaringan telekomunikasi. *Delay* merupakan waktu yang diperlukan sebuah paket untuk

melakukan perjalanan dari pengiriman ke penerima [15]. Persamaan *delay* rata – rata dapat dituliskan sebagai berikut:

$$\boldsymbol{Delay} = \frac{\text{Waktu penerimaan paket} - \text{waktu pengiriman paket}}{\text{jumlah paket yang diterima}} \ldots\ldots (\mathbf{2.2})$$

c) *Jitter*

Jitter merupakan sebagai variasi *delay* yang diakibatkan oleh panjang *quene* dalam suatu waktu pengolahan data, *reassemble* paket – paket data diakhir pengiriman akibat kegagalan sebelumnya dan proses pengiriman paket dalam media [15]. Persamaan *jitter* dapat dituliskan sebagai berikut:

$$\boldsymbol{Jitter} = \frac{\text{total variasi } delay}{\text{total paket yang diterima}} \ldots\ldots\ldots (\mathbf{2.3})$$

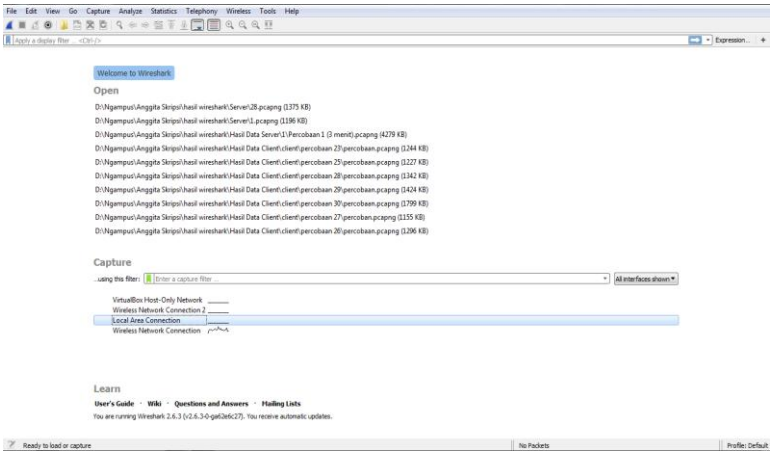
d) *Throughput*

Throughtput adalah nilai rata – rata pengiriman yang sukses melalui saluran telekomunikasi dalam suatu pengiriman. *Throughtput* diukur dalam satuan *bit per second* (bps atau bit/s) [15]. *Throughput* dapat dituliskan sebagai berikut.

$$\boldsymbol{Throughput(bps)} = \frac{\text{Paket data yang diterima}}{\text{Waktu pengiriman paket}} \ldots\ldots\ldots (\mathbf{2.4})$$

2.2.7 *Wireshark*

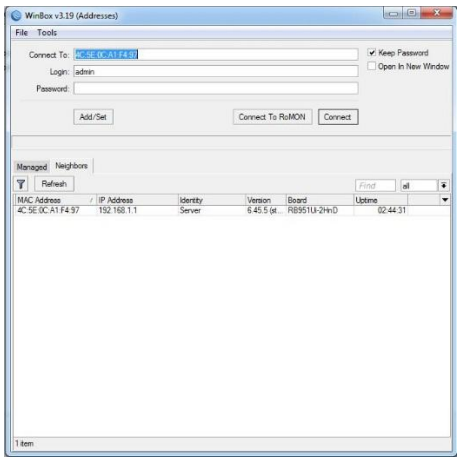
WireShark adalah *network protocol analyzer* terkemuka di dunia. Ini memungkinkan pengguna dapat melihat apa yang sedang terjadi pada jaringan pengguna pada level *microscopic*. *WireShark* sudah diakui secara standar *de facto* dan *de jure* di banyak industri dan lembaga pendidikan. Aplikasi ini menangkap paket-paket jaringan dan berusaha untuk menampilkan semua informasi di paket tersebut se detail mungkin. Hal ini dapat di ibaratkan sebagai alat untuk memeriksa apa yang sebenarnya sedang terjadi di dalam jaringan. Berikut adalah Gambar 2.7 Tampilan *WireShark* [16].



Gambar 2. 7 *WireShark.*

2.2.8 *Winbox*

Winbox adalah sebuah *software* atau *utility* yang di gunakan untuk me-*remote* sebuah *server mikrotik* ke dalam mode *Graphical User Interface* (GUI) melalui *operating system windows* berikut adalah Gambar 2.8 Tampilan *Winbox* [17].



Gambar 2. 8 Tampilan *Winbox*.