BAB II

LANDASAN TEORI

2.1. Definisi Virtual Private Network

Virtual Private Network adalah cara untuk mensimulasikan jaringan pribadi melalui jaringan publik, seperti internet. Disebut *"virtual"* karena bergantung pada penggunaan *virtual* yaitu koneksi, koneksi sementara yang tidak memiliki kehadiran fisik secara nyata, tetapi terdiri dari paket diarahkan melalui variasi mesin di internet secara *ad-hoc*. Koneksi *virtual* yang aman yang dibuat di antara dua mesin, mesin dan jaringan, atau dua jaringan (Mairs,J.2002).

Menurut *IETF, Internet Enggineering Task Force VPN* merupakan suatu bentuk *private* internet yang melalui publik network (Internet), dengan menekankan pada keamanan data dan akses global melalui internet. Hubungan ini dibangun melalui suatu *tunnel* (terowongan) *virtual* antara dua node.

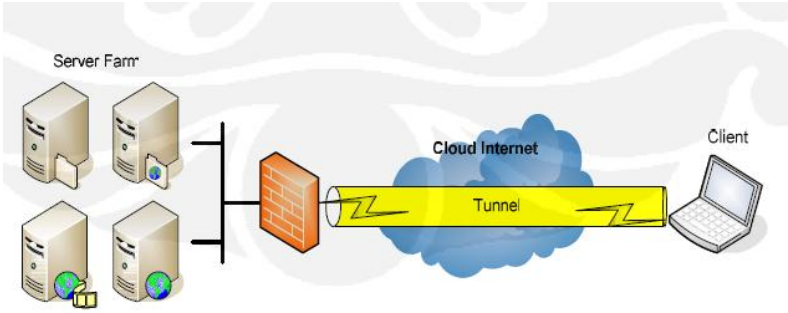
Data *dienkapsulasi* (dibungkus) dengan *header* yang berisi informasi routing untuk mendapatkan koneksi *point-to-point* sehingga data melewati jaringan publik dan dapat mencapai akhir tujuan. Sedangkan untuk mendapatkan koneksi yang bersifat private, data harus *dienkripsi* terlebih dahulu untuk menjaga kerahasiannya sehingga paket yang tertangkap ketika melewati jaringan publik tidak terbaca karena harus melewati proses *dekripsi* (Wendy,A.,Ramadhana,A.,2005)

2.2. Teknologi Tunneling

Teknologi *tunneling* merupakan teknologi yang bertugas untuk menangani dan menyediakan koneksi point-to-point dari sumber ke tujuannya. Disebut tunnel karena koneksi *point-to-point* tersebut sebenarnya terbentuk dengan melintasi jaringan umum, namun koneksi tersebut tidak mempedulikan paket-paket data milik orang lain yang sama-sama melintas jaringan umum tersebut, tetapi koneksi tersebut hanya melayani transportasi data dari pembuatnya. Koneksi *point-to-point* ini sesungguhnya tidak benar-benar ada, namun data yang dihantarkan seperti benar-benar melewati koneksi pribadi yang bersifat *point-to-point (Cisco System.2001).*

Teknologi ini dapat dibuat di atas jaringan dengan pengaturan IP Addressing dan IP *routing* yang sudah matang. Maksudnya, antara sumber *tunnel* dengan tujuan tunnel telah dapat saling berkomunikasi melalui jaringan dengan pengalamatan IP. Apabila komunikasi antara sumber dan tujuan dari *tunnel* tidak dapat berjalan dengan baik, maka *tunnel* tersebut tidak akan terbentuk.

Untuk membuat sebuah *tunnel*, diperlukan sebuah *protocol* pengaturnya sehingga *tunnel* secara logika ini dapat berjalan dengan baik bagaikan koneksi *point-to-point* sungguhan. Saat ini, tersedia banyak sekali *protocol pembuat tunnel* yang bisa di gunakan seperti PPTP dan L2TP.



Sumber : *www.google.co.id/search?q=tunneling*

Gambar 2.2 Topologi tunneling

2.3. Point to point tunneling protocol (PPTP)

PPTP merupakan protokol jaringan yang memungkinkan pengamanan *transfer* data dari *remote client* ke *server* pribadi perusahaan dengan membuat sebuah VPN melalui TCP/IP.

Teknologi jaringan PPTP merupakan pengembangan dari *remote access point-to-point protocol* yang dikeluarkan oleh *Internet Enggineering Task Force (IETF)*. PPTP merupakan protokol jaringan yang merubah paket PPP menjadi IP datagrams agar dapat ditransmisikan melalui internet *(Cisco System.2001)*.

Fasilitas utama dari penggunaan PPTP adalah dapat digunakannya *public switched telephone network (PSTNs)* untuk membangun VPN. Pembangunan PPTP yang mudah dan berbiaya murah untuk digunakan secara luas, menjadi solusi untuk *remote users* dan *mobile users* karena PPTP memberikan keamanan dan *enkripsi* komunikasi melalui PSTN ataupun Internet.

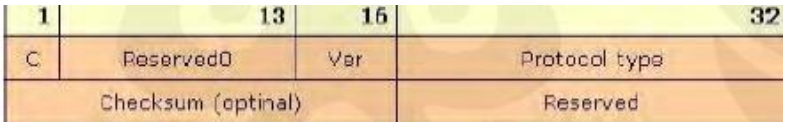
2.3.1 Generic Routing Encapsulation (GRE)

Generic Routing Encapsulation (GRE) adalah *tunneling* protokol yang dikembangkan oleh *cisco*, protokol ini dapat melakukan enkapsulasi berbagai

macam jenis paket dalam lapisan *network* protokol dalam *tunnelnya*, dengan cara membuat *virtual* komunikasi *point to point* dari router asal ke router tujuan dengan menggunakan IP pada komunikasi *internetwork (www.cisco.com).*

Generic Routing Encapsulation (GRE) protokol *tunneling* yang memiliki kemampuan membawa lebih dari satu jenis protokol pengalamatan komunikasi. Bukan hanya paket beralamat IP yang dapat dibawanya, melainkan banyak protokol lain seperti CNLP, IPX, dan banyak lagi *(Cisco System.2001).* kemudian, semua itu dibungkus atau dienkapsulasi menjadi sebuah paket yang bersistem pengalamatan IP. Paket tersebut didistribusikan melalui system *tunnel* yang juga bekerja di atas protokol komunikasi IP. Dengan menggunakan *tunneling* GRE, router yang ada pada ujung *tunnel* melakukan enkapsulasi paket-paket protokol lain di dalam *header* dari protokol IP. Hal ini akan membuat paket-paket tadi dapat dibawa ke manapun dengan cara dan metode yang terdapat pada teknologi IP. Dengan adanya kemampuan ini, maka protokol-protokol yang dibawa oleh paket IP tersebut dapat lebih bebas bergerak ke manapun lokasi yang ditujuh, asalkan terjangkau secara pengalamatan IP.

Generic Routing Encapsulation (GRE) banyak digunakan untuk memperpanjang dan mengekspansi jaringan lokal yang dimiliki si penggunanya. Meskipun cukup banyak digunakan, *Generic Routing Encapsulation* (GRE) juga tidak menyediakan *system enkripsi* data yang lalu lalang di *tunnel*-nya, sehingga semua aktivitas datanya dapat dimonitor menggunakan protokol *analyzer*.



Sumber: http://packetlife.net/blog/2012/feb/27/gre-vs-ipip-tunneling

Gambar 2.3 Format Header GRE

Keterangan:

- C : Checksum Present
- Reserved 0&1 : Disediakan untuk digunakan kemudian
- Ver : Version number; harus – 0.
- Protocol Type : Berisi *protocol type* dari *payload packet.*
- Checksum : Berisi IP checksum dari header GRE dan *payload packet.*

2.3.2 Artsitektur PPTP

Komunikasi yang aman dibuat dengan menggunakan protokol PPTP melewati tiga proses, dimana setiap proses tersebut membutuhkan selesainya proses yang sebelumnya. Ketiga proses tersebut berjalan dengan cara berikut:

- PPTP *Connection and Communication*. Klien PPTP menggunakan PPP untuk terhubung Ke ISP. Koneksi tersebut menggunakan protokol PPP untuk membangun koneksi dan enkripsi paket data.
- PPTP *Control Connection*. Menggunakan koneksi ke internet yang telah dibangun oleh protokol PPP, protokol PPTP membuat sebuah control connection dari klien PPTP ke server PPTP di internet. Koneksi tersebut menggunakan TCP untuk membangun koneksi dan ini disebut dengan PPTP tunnel.

- PPTP Data *Tunneling*. Akhirnya protokol PPTP membuat IP *datagrams* yang didalamnya terdapat *enkripsi* paket PPP yang kemudian dikirim melalui PPTP tunnel ke server PPTP, server PPTP membongkar IP datagram dan *mendekripsi* paket PPP dan kemudian merutekan paket yang telah *didekripsi* ke jaringan *private.*

PPTP Control Connection

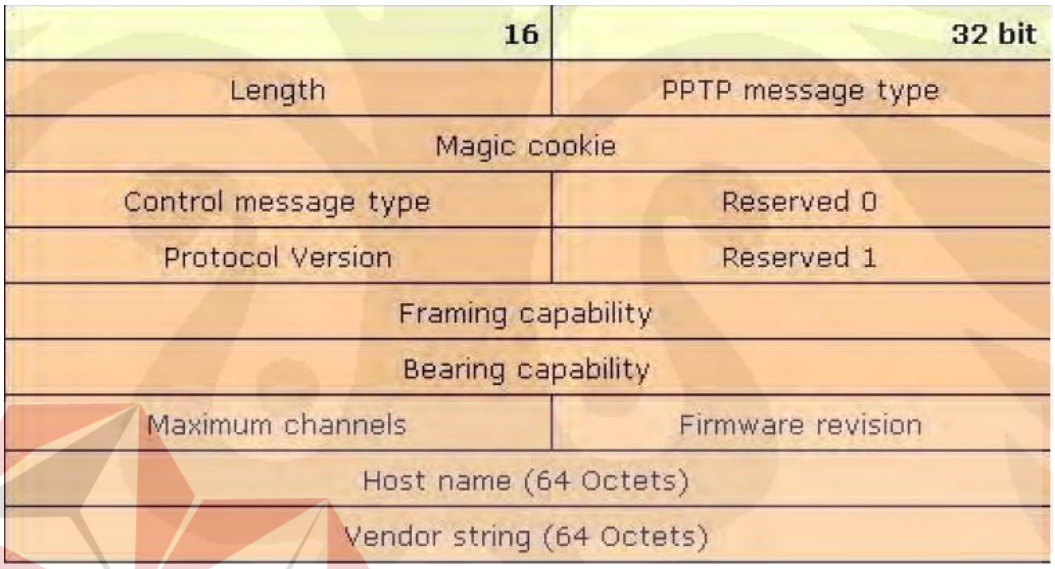
adalah protokol PPTP yang menspesifikasikan seri pengiriman dari *control message* antara PPTP-*enabled client* dan *server* PPTP. *Control message* membangun, memelihara dan mengakhiri PPTP *tunnel*. Berikut ini merupakan daftar yang dibuat oleh control message dasar yang digunakan untuk membuat dan memelihara PPTP *tunnel:*

- PPTP_START_SESSION_REQUEST : Permintaan untuk memulai session
- PPTP_START_SESSION_REPLY : untuk menjawab start session
- PPTP_ECHO_REQUEST : maintain session
- PPTP_ECHO_REPLY : untuk menjawab maintain session
- PPTP_WAN_ERROR_NOTIFY : Laporan error pada koneksi PPP
- PPTP_SET_LINK_INFO : merubah setting koneksi antara klien dan server PPTP
- PPTP_STOP_SESSION_REQUEST : Mengakhiri session
- PPTP_STOP_SESSION_REPLY : Untuk menjawab stop session

Control message ditransmisikan pada paket control pada TCP datagram. Satu koneksi TCP dibangun antara klien PPTP dan server PPTP. Koneksi tersebut digunakan untuk menukar control message. Control message dikirim dengan TCP

datagram. Penukaran message antara klien PPTP dan server PPTP melalui koneksi TCP digunakan untuk membuat dan memelihara PPTP *tunnel*.

2.3.3 Format Header PPTP



16	32 bit
Length	PPTP message type
Magic cookie	
Control message type	Reserved 0
Protocol Version	Reserved 1
Framing capability	
Bearing capability	
Maximum channels	Firmware revision
Host name (64 Octets)	
Vendor string (64 Octets)	

Sumber: http://www.faqs.org/rfcs/rfc2637.html

Gambar 2.3.3 Header PPTP

Keterangan :

- Length : Panjang total paket PPTP dalam octet termasuk header PPTPnya.
- PPTP message type : tipe message; 1 control message, 2 management message
- Magic cookie : magic cookie selalu terkirim 0x1A2B3C4D. untuk mengizinkan receiver menjamin sinkronisasi dengan TCP data stream.
- Control Message Type :
  - Control connection management – 1 start-control-connection-request; 2 start-control-connection-reply; 3 stop-control-

connection-request;4 stop-control-connection-reply; 5 echo-request; 6 echo-reply.

- Call management – 7 Outgoing-call-request; 8 outgoing-call-reply; 9 Incoming-call-requst; 10 incoming-call-reply; 11 incoming-call-connected; 12 call-clear-request; 13 call-disconnect-notify
- Error reporting – 14 wan-error-notify
- PPP session control – 15 Set-link-info

- Reserved 0 & 1 : Harus = 0
- Protocol version : PPTP version number
- Framming Capabilities : Mengindikasikan tipe framing yang dapat dilakukan oleh pengirim : 1 – Asynchronous Framming supported; 2 – Synchronous Framming supported
- Bearer Capabilities : mengindikasikan kemampuan bearer yang dapat dilakukan oleh pengirim: 1 - Analog access supported; 2 – digital access supported
- Maximum Channels : Jumlah total session PPP yang dapat didukung PAC.
- Firmware Revision : berisi jumlah firmware revision dari PAC jika dikeluarkan oleh PAC atau versi dari PNS PPTP jika dikeluarkan oleh PNS
- Host Name : berisi nama DNS dari PAC atau PNS

- Vendor Name : berisi string vendor tertentu menjelaskan tipe PAC yang digunakan, atau tipe software PNS yang digunakan jika request dikeluarkan oleh PNS.

2.4. Layer 2 Tunneling Protocol (L2TP)

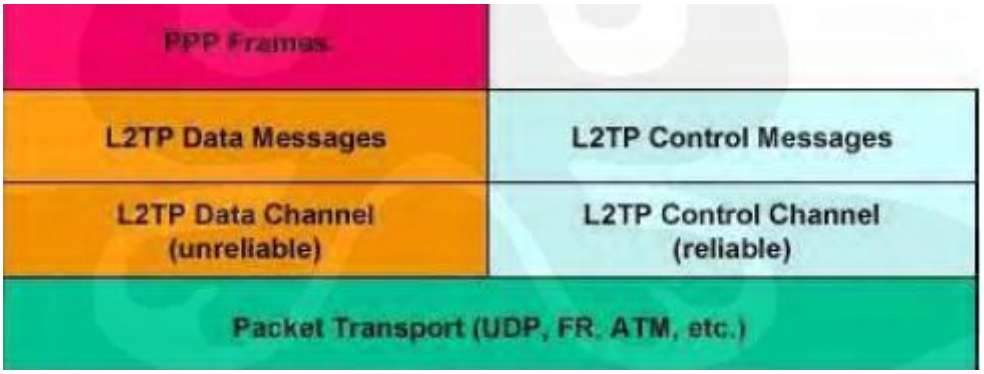
L2TP adalah sebuah *tunneling protocol* yang memadukan dan mengombinasikan duah buah *tunneling protocol* yaitu *L2F (Layer 2 Forwarding)* milik *cisco* systems dengan PPTP (Point-to-Point Tunneling Protocol) milik *Microsoft* (http://www.faqs.org/rfcs/rfc2637.html).

Pada awalnya, semua produk cisco menggunakan *L2F* untuk mengurus tunneling-nya, sedangkan *operating system Microsoft* yang terdahulu hanya menggunakan PPTP untuk melayani penggunanya yang ingin bermain dengan *tunnel*. Namun saat ini, Microsoft windows NT/2000 telah dapat menggunakan PPTP atau L2TP dalam teknologi VPN-nya

L2TP biasanya digunakan dalam membuat *virtual private dial network* (VPDN) yang dapat bekerja membawa semua jenis protokol komunikasi didalamnya. Selain itu, L2TP juga bersifat media independen karena dapat bekerja di atas media apapun. L2TP memungkinkan penggunanya untuk tetap dapat terkoneksi dengan jaringan local milik mereka dengan *policy* keamanan yang sama dan dari manapun mereka berada, melalui koneksi VPN atau VPDN.

Protokol L2TP sering disebut sebagai protokol dial-up virtual, karena L2TP memperluas suatu session PPP (point-to-point protocol) dial-up melalui jaringan *public internet*, atau sering juga digambarkan seperti koneksi virtual PPP.

2.4.1. *Arsitektur L2TP*



Sumber: http://www.faqs.org/rfcs/rfc2637.html

Gambar 2.4 Struktur Protokol L2TP

Frame PPP *dienkapsulasi* oleh *header* L2TP dan paket transport UDP, kemudian dilewatkan melalui data *channel* yang *unreliable*. *Control messages* dikirimkan melalui suatu control channel L2TP yang juga mentransmisikan paket in-band melalui paket transport yang sama. *Sequence number* diperlukan pada semua *control message* dan digunakan untuk menyediakan pengiriman yang handal dalam *control channel*. Data message juga harus menggunakan *sequence number* untuk menyusun kembali dan mendeteksi paket yang hilang.

Ada 2 jenis messages yang digunakan L2TP: control messages dan data messages (http://www.faqs.org/rfcs/rfc2637.html):

1. Control messages
    - Digunakan untuk :- Establishment (pembentukan)
      - Maintanance (pemeliharaan)
      - Pemutusan tunnel L2TP dan interkoneksi
    - Menggunakan suatu control channel yang reliable didalam L2TP untuk menjamin kepastian paket yang terkirim

2. Data Messages

    - Digunakan untuk mengenkapsulasi frame PPP yang akan dibawa melalui tunnel
    - Jika loss packet terjadi, data messages tidak akan dikirim kembali (not reliable).

Tipe Control Message

Didalam protokol tunnel, control message dipertukarkan secara inband antara client dan server. Control koneksi bertanggung jawab untuk pembentukan pemutusan, dan maintenance session, yang dibawa dalam tunnel dan tunnel itu sendiri

Tipe control message adalah sebagai berikut:

- Control Connection Management :
    - 0 = (reserved)
    - 1 = (SCCRQ) Start-Control-Connection-Request
    - 2 = (SCCRP) Start-Control-Connection-Reply
    - 3 = (SCCCN) Start-Control-Connection-Connected
    - 4 = (StopCCN) Start-Control-Connection-Notification
    - 5 = (reserved)
    - 6 = (HELLO) Hello
- Call Management
    - 7 = (OCRQ) Outgoing-Call-Request
    - 8 = (OCRO) Outgoing-Call-Reply
    - 9 = (OCCN) Outgoing-Call-Connected

- 10 = (ICRQ) Incoming-Call-Request
- 11 = (ICRP) Incoming-Call-Reply
- 12 = (ICCN) Incoming-Call-Connected
- 13 = (reserved)
- 14 = (CDN) Call-Disconnect-Notify

➢ Error Reporting

- 15 = (WEN) WAN-Error-Notify

➢ PPP Session Control

- 16 = (SLI) Set-Link-Info

Definisi control message di atas adalah sebagai berikut :

- SCCRQ - control message yang digunakan untuk menginisialisasi tunnel antara server dan client, dikirim oleh client dan server untuk proses pembentukan tunnel.
- SCCRP - Control message yang digunakan untuk mengindikasikan bahwa SCCRQ telah diterima dan pembentukan tunnel harus dilanjutkan. Dikirim sebagai jawaban dari SCCRP.
- StopCNN – Control messages yang dikirim oleh client dan server untuk menginformasikan peer bahwa tunnel sedang diputus dan hubungan control harus diputus. Lebih lanjut lagi seluruh koneksi akan terputus (tanpa mengirim explicit call control message).
- OCRQ – control message yang dikirim oleh server ke client untuk mengindikasikan bahwa outbound call dari client

terbentuk. Merupakan message pertama dalam pertukaran message yang digunakan untuk membentuk session dalam tunnel L2TP.

- OCRP – control message yang dikirim oleh client kepada server sebagai respon OCRQ yang dikirim. Merupakan message kedua yang bertukar pada pembentukan session dalam tunnel L2TP.
- OCCN – control message yang dikirimkan client ke server mengikuti OCRP setelah outgoing call terbentuk. Merupakan message terakir yang bertukar untuk pembentuan session dalam tunnel L2TP. OCCN digunakan juga untuk mengindikasikan hasil dari permintaan outgoing call yang berhasil dan memberikan informasi pada server mengenai parameter yang diperoleh setelah panggilan terbentuk seperti tipe message, (TX) connection speed, dan tipe framing.

2.4.2.Format Header L2TP

Sumber: http://www.faqs.org/rfcs/rfc2637.html

Gambar 2.4.2 Format Header L2TP

Keterangan :

- Type (T) bit : - tipe dari message
  - 0 = data message, 1 = control message
- Length (L) bit : - L = 1, berarti field length terisi
  - untuk control message harus di set = 1
- X bit : - disediakan untuk digunakan kemudian
  - semua bit yang dipesan harus di set 0 pada outgoing message dan pada incoming message diabaikan
- Sequence (S) bit : - S = 1, berarti field NS dan Nr terisi
  - untuk control message harus di set = 1
- Offset (O) bit : - O = 1, Field size offset terisi
  - untuk control message di set = 0 (Nol)
- Priority (P) bit : - P = 1, mendapatkan perlakuan yang istimewa khususnya dalam data message
  - Untuk semua control message di set = 0
- Ver : - Ver = 2, versi header data message L2TP
  - Jika unknown ver, paket tersebut harus dibuang.
- Length Field : - panjang total dari message (byte).
- Tunnel ID : - Identifier untuk control connection
  - Significant local saja
- Session ID : - identifier untuk suatu session di dalam

suatu tunnel

- Significant local saja

- NS Sequence Number : - Sequence number untuk control message
- Nr Sequence Number : - Sequence number control message berikutnya yang diterima.
- Offset Field : - start dari payload data.

2.5. Quality of Service

Quality of Service (QoS) adalah kemampuan suatu jaringan untuk menyediakan layanan yang baik dengan menyediakan kapasitas jaringan, mengatasi *packet loss, delay* dan *throughput* (Langi, 2011). Sedangkan menurut Rahayu, (2013) kualitas layanan atau QoS adalah kemampuan sebuah jaringan untuk menyediakan layanan yang lebih baik bagi trafik. QoS merupakan sebuah sistem arsitektur *end-to-end* dan bukan merupakan sebuah *feature* yang dimiliki oleh jaringan. QoS suatu *network* merujuk ke tingkat kecepatan dan keandalan penyampaian berbagai jenis beban data di dalam suatu komunikasi.

QoS dirancang untuk membantu pengguna menjadi lebih produktif dengan memastikan bahwa pengguna mendapatkan kinerja yang handal dari aplikasi – aplikasi berbasis jaringan. QoS mengacu pada kemampuan jaringan untuk menyediakan layanan yang lebih baik pada trafik jaringan tertentu melalui teknologi yang berbeda – beda. QoS merupakan suatu tantangan yang besar dalam jaringan berbasis IP dan internet secara keseluruhan (Langi, 2011)

QoS dapat dilihat dari tingkat kecepatan dan keandalan dalam mengelola penyampaian data dalam suatu informasi dengan jenis beban yang beragam. Terdapat beberapa parameter yang digunakan untuk mengukur tingkat kecepatan

dan keandalan satu jaringan, diantaranya *latency* (*delay*), *packet loss* dan *troughput.*

2.6. Parameter – parameter *Quality of Service*

QoS mempunyai beberapa parameter namun berikut adalah parameter – parameter yang digunakan:

2.6.1. Delay

Delay merupakan akumulasi berbagai waktu tunda dari ujung ke ujung pada jaringan. Waktu tunda mempengaruhi waktu tempuh paket untuk mencapai tujuan (Langi, 2011).

2.6.2. Packet Loss

Paket hilang (*packet loss*) merupakan penyebab utama pelemahan audio dan video pada multimedia *streaming*. Paket hilang dapat disebabkan oleh pembuangan paket di jaringan (*network loss*) atau pembuangan paket di *gateway/terminal* sampai kedatangan terakhir (*late loss*). *Network loss* secara normal disebabkan kemacetan (*router buffer overflow*), perubahan rute secara seketika, kegagalan *link*, dan *lossy link* seperti saluran *wireless*. Kemacetan atau kongesti pada jaringan merupakan penyebab utama dari paket hilang (Langi, 2011).

2.6.3. Throughput

Throughput merupakan *rate* (kecepatan) transfer data efektif, yang diukur dalam *bit per second* (bps). *Throughput* merupakan jumlah total kedatangan paket yang sukses yang diamati pada *destination* selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut (Langi, 2011).

2.7. Definisi alamat *Internet Protocol*

Alamat *internet protocol* (IP) terdiri atas 32 bit angka, pada umumnya ditulis dalam notasi *dotted-decimal*. "*Decimal*" merupakan istilah yang berasal dari setiap *byte* (8 bit) pada 32 bit alamat IP yang di konversi kedalam desimal. Dari keempat angka desimal yang dihasilkan tertulis didalam urutan, dengan "*dots*," atau titik yang memisahkannya dinamakan *dotted-decimal*. Setiap angka decimal pada alamat IP disebut *octet*. Istilah *octet* digunakan secara umum bukan *byte*. Ukuran angka desimal disetiap oktetnya berkisar antara 0 hingga 255. (Odom, 2004).

2.7.1. Jenis Alamat

Pada tulisan memorandum yang ditulis oleh insinyur dan ilmuan komputer tentang metode, perilaku, penelitian, atau inovasi yang berlaku untuk kinerja internet dan sistem yang tersambung ke internet (RFC 790) mendefinisikan protokol IP, termasuk beberapa perbedaan kelas dari sebuah jaringan. IP didefinisikan kedalam tiga bagian kelas jaringan yang berbeda yaitu A, B, dan C, yang digunakan oleh *host*.

2.7.2. Kelas IPv4

Setiap jaringan kelas A, B, dan C mempunyai perbedaan ukuran sebagai identifikasi jaringan :

1. **Kelas A** adalah alamat jaringan yang mempunyai panjang 1 *byte* untuk jaringan. 3 *bytes* sisanya untuk bagian *host*.
2. **Kelas B** adalah alamat jaringan yang mempunyai panjang 2 *bytes* untuk jaringan. 2 *bytes* sisanya untuk bagian *host*.

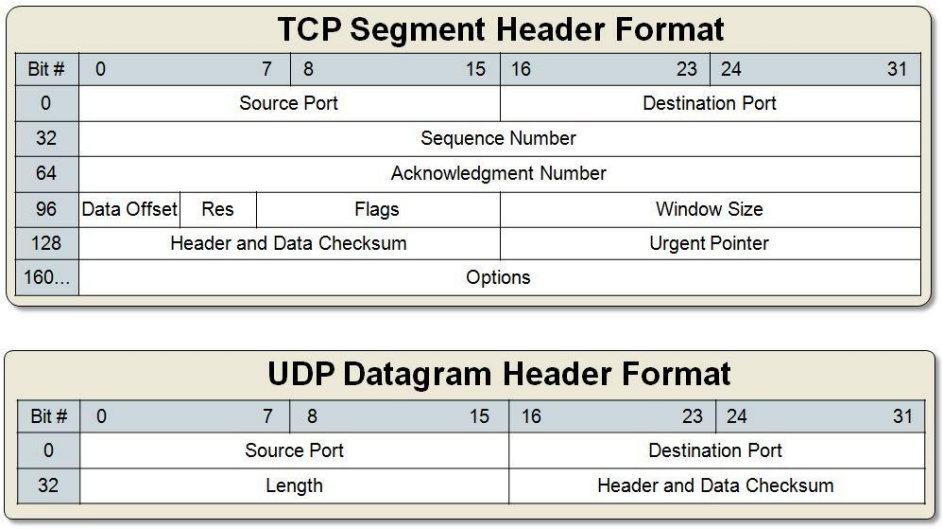
3. **Kelas C** adalah alamat jaringan yang mempunyai panjang 3 *bytes* untuk jaringan. 1 *bytes* sisanya untuk bagian *host*.

2.8. *User Datagram Protocol* (UDP)

UDP menyediakan layanan aplikasi untuk saling bertukar pesan. Tidak seperti TCP, UDP merupakan *connectionless*, *no reliability*, *no windowing*, dan tanpa melakukan penataan kembali data yang diterima. Akan tetapi UDP memberikan beberapa fungsi dari TCP, seperti pengiriman data, segmentasi, dan *multiplexing* yang menggunakan angka *port*, dan juga melakukan dengan *byte* lebih sedikit dari yang disediakan dan sedikit pemrosesan.

Multiplexing pada UDP akan menggunakan angka *port* untuk identitas sama seperti pada TCP. Satu – satunya perbedaan dalam soket UDP bahwa, sebagai gantinya menunjuk seperti halnya protokol *transport* pada TCP, UDP adalah protokol *transport*. Suatu aplikasi dapat membuka identitas angka *port* pada *host* yang sama namun menggunakan TCP dalam satu kasus dan disisi lain menggunakan UDP itu jarang terjadi, tapi hal tersebut tentunya diperbolehkan. Jika suatu layanan tertentu mendukung *transport* UDP atau TCP, akan menggunakan nilai *port* yang sama angka port TCP dan UDP.

Data transfer UDP berbeda dengan data transfer pada TCP bahwa tidak ada penataan kembali. Penggunaan aplikasi UDP mentoleransi terjadinya kehilangan data, atau mempunyai suatu mekanisme untuk mendapatkan kembali data yang hilang.



TCP Segment Header Format

Bit #	0 – 7	8 – 15	16 – 23	24 – 31
0	Source Port		Destination Port	
32	Sequence Number			
64	Acknowledgment Number			
96	Data Offset	Res	Flags	Window Size
128	Header and Data Checksum		Urgent Pointer	
160...	Options			

UDP Datagram Header Format

Bit #	0 – 7	8 – 15	16 – 23	24 – 31
0	Source Port		Destination Port	
32	Length		Header and Data Checksum	

Sumber : https://microchip.wdfiles.com

Gambar 2.5 *Header* TCP dan UDP.

Pada Gambar 2.5 menunjukkan format *header* dari TCP dan UDP. Perhatikan kedua *source port* dan *destination port* pada *header* TCP dan UDP, pada UDP tidak ada *sequence number* dan *acknowledgement*. UDP tidak membutuhkan bagian tersebut karena hal tersebut membuatnya tidak adanya penomoran data untuk *acknowledgements* atau *sequencing*.

UDP mempunyai beberapa keunggulan dibandingkan TCP dengan tidak adanya *acknowledgement* dan *sequence*. Keuntungan yang paling jelas dari UDP adalah memiliki lebih sedikit *byte* dari yang disediakan. Tidak jelas seperti sebenarnya UDP tidak perlu menunggu *acknowledgement* atau menahan data di memori hingga setelah *acknowledgment*. Dengan demikian aplikasi UDP tidak diperlambat dengan proses *acknowledgement*, dan memorinya terbebas sehingga lebih cepat (Odom, 2004).

2.9. Mikrotik

Mikrotik adalah sistem operasi dan perangkat lunak yang dapat digunakan untuk menjadikan komputer menjadi *router network* yang handal, mencakup

berbagai fitur yang dibuat untuk IP *network* dan jaringan *wireless*, cocok digunakan oleh ISP dan *provider hotspot*. Untuk instalasi Mikrotik tidak dibutuhkan piranti lunak tambahan atau komponen tambahan lain. Mikrotik didesain untuk mudah digunakan dan sangat baik digunakan untuk keperluan administrasi jaringan komputer seperti merancang dan membangun sebuah sistem jaringan komputer skala kecil hingga yang kompleks sekalipun (Sinaga, 2013).

2.9.1. *Router* Mikrotik

Router Mikrotik mempunyai produk *routerboard* yang kecil dan diperuntukkan untuk di dalam rumah. Memiliki 4 buah *port ethernet* 10/100, dengan prosesor baru Atheros 400MHz.

Gambar 2.6 Router Mikrotik 941 *haplite*

2.10. Layanan RTSP dan RTP

Real time transmission protocol (RTP) merupakan protokol standar internet untuk pengiriman data *real time*, termasuk audio dan video. Protokol ini dapat digunakan untuk media *on demand* dan juga layanan interaktif seperti telepon internet. RTP telah dikembangkan oleh *Internet Engineering Task Force* (IETF) dan digunakan secara luas. Sebenarnya standar RTP mendefinisikan sepasang protokol yaitu RTP dan *real time transport control protocol* (RTCP).

RTP digunakan untuk pertukaran data multimedia, selama RTCP mengontrol sebagian dan digunakan secara periodik termasuk mengontrol *feedback* informasi mengenai kualitas transmisi yang berhubungan dengan data flow. RTP berjalan diatas protokol UDP/IP namun upaya yang dilakukan membuatnya menjadi *transport independence* sehingga hal tersebut seharusnya digunakan diatas protokol lain. RTP yang berhubungan dengan RTCP menggunakan *port transport layer* secara berturut – turut, ketika digunakan pada UDP.

Internet merupakan sebuah sistem yang terdiri atas komputer-komputer yang didesain untuk dapat berbagi sumber daya, berkomunikasi, dan dapat mengakses informasi. Tujuannya agar setiap bagian dari jaringan komputer dapat meminta dan memberikan layanan. Ada beberapa layanan untuk media pengiriman seperti *real time streaming protocol* (RTSP).

Seperti yang telah dideskripsikan oleh RFC 2326, pada *layer* aplikasi protokol RTSP memungkinkan untuk mengontrol melalui data yang dikirimkan dengan *real time* dari sebuah IP. Termasuk seperti mengontrol *pausing playback*, memposisikan *playback*, mempercepat atau mengembalikan *playback*. RTSP bukan bertipe mengirimkan media secara terus – menerus, meskipun demikian RTSP menyisipkan media *streaming* secara terus - menerus dengan sebisa mungkin mengendalikan *streaming*.

RTSP adalah protokol presentasi multimedia antar *client* dan *server*. Sehingga tidak ada *notion* pada koneksi RTSP. Sebagai gantinya, server mengelola identifikasian sesi label. Pada sesi RTSP protokol *transport* tidak terikat. Selama sesi RTSP terjadi, RTSP *client* akan membuka dan menutup agar koneksi pada *transport reliable* untuk *request* RTSP kepada *server*. Hal tersebut

mungkin sebagai alternatifnya menggunakan protokol transport *connectionless* seperti UDP.

RTSP didesain untuk bekerja dengan protokol tingkat dasar seperti *real time protocol* (RTP) atau *resource reservation protocol* (RSVP) untuk memberikan servis *streaming* secara komplit pada internet. Hal tersebut berarti untuk memilih kanal pengiriman (seperti UDP, *multicast* UDP dan TCP), dan mekanisme pengiriman berdasarkan RTP. Pesan RTSP dikirimkan melalui pita media *streaming*. RTSP bekerja untuk *multicast* audien yang besar seperti halnya *single viewer unicast* (Durresi, 2005).

2.11. Network Monitoring

Monitoring jaringan dibutuhkan untuk melakukan pengawasan pada jaringan yang dilakukan, agar jaringan tersebut selalu terkontrol dan apabila terputus dapat diketahui langsung oleh *user*. Pada tugas akhir ini *software* yang digunakan untuk *monitoring* jaringan yaitu Wireshark.

2.11.1. Wireshark

Wireshark merupakan salah satu *tool monitoring* jaringan yang berfungsi untuk mengawasi lalu lintas pada jaringan komputer dan dapat menganalisa keseluruhan jaringan *computer* (Cahyaningtyas, 2013). Logo wireshark dapat dilihat pada Gambar 2.6

Sumber: http://www.wireshark.org

Gambar 2.7 Logo Wireshark

Wireshark dapat melihat dan meyimpan informasi mengenai paket keluar dan

masuk dalam jaringan yang terkirim dan diterima.

2.11.2. Tujuan dan Manfaat Wireshark

Manfaat dari *software* Wireshark, sebagai berikut :

- Menangkap informasi yang dikirim dan diterima,
- Mengetahui aktivitas dalam jaringan komputer,
- Mengetahui dan menganalisa kinerja jaringan computer,
- Mengamati keamanan jaringan komputer (Cahyaningtyas, 2013).