BAB II

LANDASAN TEORI

2.1 VoIP (*Voice over Internet Protocol*)

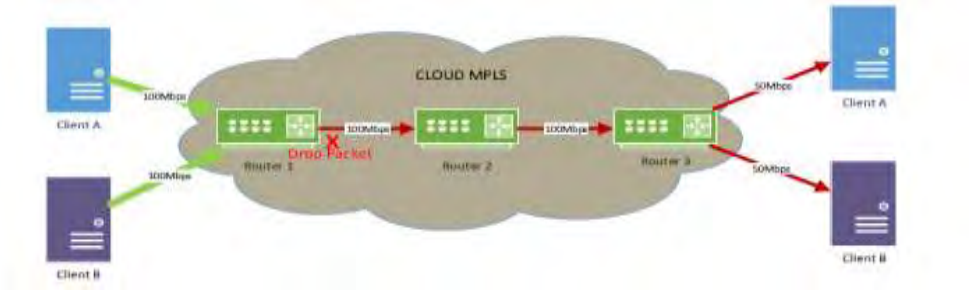
VoIP merupakan teknologi yang memfasilitasi untuk melakukan panggilan telepon melalui jaringan IP (*Internet Protocol*) berbentuk paket secara *real-time*. VoIP merupakan singkatan dari *Voice over Internet Protocol* dan bisa dapat disebut juga sebagai telepon *internet*, atau telepon *broadband* (Nugroho, 2010). Teknik dasar *Voice over Internet Protocol* atau yang biasa dikenal dengan sebutan VoIP adalah teknologi yang memungkinkan kemampuan melakukan percakapan telepon dengan menggunakan jalur komunikasi data pada suatu jaringan (*networking*). Sehingga teknologi ini memungkinkan komunikasi suara menggunakan jaringan berbasis IP (*internet protokol*) untuk dijalankan diatas infrastruktur jaringan *packet network*. VoIP melewatkan trafik suara, video dan data yang berbentuk paket secara *real-time* dengan jaringan *Internet Protocol*. VoIP ini dapat memanfaatkan infrastruktur internet yang sudah ada untuk berkomunikasi seperti layaknya menggunakan telepon biasa dan tidak dikenakan biaya telepon biasa untuk berkomunikasi dengan pengguna VoIP lainnya dimana saja dan kapan saja. Jaringan yang digunakan data, dikirimkan dan dipulihkan kembali dalam bentuk *voice* dipenerima. *Voice* diubah dulu ke dalam format digital karena lebih mudah dikendalikan dalam hal ini dapat dikompresi, dan dapat diubah ke format yang lebih baik dan data digital lebih tahan terhadap *noise* daripada analog (Hidayat, 2019).

2.2 *Network Congestion Control*

Congestion control meliputi perancangan mekanisme dan algoritma pada keterbatasan kapasitas atau pengaturan sumber daya trafik secara dinamis. Itu

dapat dilakukan dengan solusi statis seperti penambahan memori *buffer*, menyediakan *link* yang lebih cepat atau dengan prosesor yang cepat, itu semua tidak efektif dalam pengaturan kemacetan. Penggunaan *internet* didominasi oleh trafik TCP seperti *Telnet*, FTP, *Web traffic* dan email. Layanan TCP ini adalah 90% pada keseluruhan trafik di *internet* dengan 50-70% trafik ini *short-lived connection* dalam ukuran dan *lifetime*. Meskipun aplikasi-aplikasi ini lebih elastis, tetapi ada hubungannya dengan *packet delay* atau *paket losses* yang akan menjadi macet dan masalah yang serius terjadi congestion (Fajri, 2016). *Congestion* pada lalu lintas jaringan disebabkan oleh (Nurlinaamik, 2014):

a. Terlalu banyak *host* dalam sebuah *broadcast domain*, *host* adalah alat yang terhubung ke jaringan dan dapat menerima dan mengirimkan informasi dari alat ke alat lainnya dalam jaringan tersebut. *Broadcast domain* adalah kumpulan dari alat-alat pada sebuah segmen jaringan yang menerima paket *broadcast* yang dikirim oleh alat lain dalam segmen jaringan tersebut.
b. *Broadcast Storm,* terjadi karena semua alat mengirimkan paket *broadcast* ke seluruh alat lain melalui jaringan. Semakin banyak *host* maka semakin besar *broadcast storm*.
c. *Multicasting,* jika dalam satu jaringan terdapat banyak komputer di mana setiap komputer mengakses beberapa halaman situs *web* bervolume tinggi dalam satu waktu yang sama, maka besar kemungkinan akan terjadi kemacetan bandwidth. Jalur yang kecil akan membuat lalu lintas jaringan padat jika dilewati oleh banyak data dalam satu periode.
d. Data *Collision*, yaitu suatu kondisi *network* dimana sebuah alat mengirimkan paket data ke sebuah segmen jaringan yang kemudian memaksa semua alat lain yang ada di segmen jaringan tersebut untuk memperhatikan paketnya. Pada saat yang bersamaan alat yang berbeda mencoba untuk mengirimkan paket yang lain, yang mengakibatkan tabrakan (*collision*), paket yang dikirim menjadi rusak akibatnya semua alat harus melakukan pengiriman ulang paket.
e. *Bandwidth* yang kecil, media jaringan dengan *bandwith* kecil tidak seimbang dengan banyaknya lalu lintas data yang terjadi sehingga dapat mengakibatkan *overload*.



Gambar 2.1 Kondisi *Network Congest*

2.3 MPLS (*Multi Protocol Label Switching*)

MPLS adalah teknologi *label-switching* yang menggabungkan kemampuan rekayasa trafik ATM dengan fleksibilitas dan skalabilitas jaringan IP. MPLS memiliki kemampuan membentuk *tunnel* atau *virtual circuit* yang melintasi *network*. Prinsip kerjanya mengabungkan beberapa keuntungan dari sistem komunikasi *circuit-switched* dan *packet-switched*. MPLS memiliki arsitektur *packet switching* dan *routing* yang sangat baik dan dapat mengirimkan data dengan sangat cepat (Soewito *et al*,. 2017). *Multi Protocol Label Switching* (MPLS) adalah teknologi penerusan paket yang banyak digunakan di penyedia layanan dan jaringan perusahaan, yang menggunakan label untuk membuat keputusan *forwarding*. Dibandingkan dengan jaringan IP tradisional, di mana router menjalankan beberapa protokol routing membuat keputusan *forwarding* independen berdasarkan *header* paket IP. Setiap *router* menganalisis *header*, terlihat pada tabel routingnya sendiri dan memilih *hop* berikutnya yang benar. Pencarian ini harus dilakukan secara independen setiap kali pada setiap paket IP tunggal, karena isi dari tabel *routing* dapat berubah sesekali (Dian *et al*,. 2017).

2.4 *Quality of Service* (QoS)

QoS adalah hasil kolektif dari berbagai kriteria performansi (parameter) yang menentukan tingkat kepuasan penggunaan suatu layanan. Umumnya QoS dikaji dalam kerangka pengoptimalan kapasitas *network* untuk berbagai jenis layanan, tanpa terus menerus menambah dimensi *network* (ITU-T, 2001). QoS adalah

kemampuan untuk memberikan jaminan *resources* dan diferensasai layanan dalam jaringan (Othman *et al*,. 2012). Prinsip kerjanya memberikan batasan-batasan tertentu dalam sebuah layanan dalam jaringan. Terdapat beberapa parameter untuk mengukur kualitas QoS diantaranya *troughput, packet loss* dan *jitter*. QoS saat ini menjadi parameter utama untuk mengukur kehandalan dari jaringan komputer, semakin tinggi nilai QoS yang didapat maka semakin handal jaringan komputer tersebut. Pada beberapa tahun sebelumnya teknologi pengiriman data ATM sangat banyak digunakan untuk mengirim data mendapat nilai QoS yang tinggi (Nurhaida & Ngadiyono, 2019) namun terlalu rumit dan mahal untuk mengimplementasikan teknologi tersebut.

2.4.1 *Packet Loss*

Packet Loss merupakan suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang dapat terjadi karena *collision* dan *congestion* pada jaringan (Nindya *et al*,. 2017). Tujuan pengukuran *packet loss* dilakukan untuk melihat kehandalan metode yang digunakan dalam mengiriman paket pada saat terjadi *network*-congestion. Satuan yang dipakai adalah persen (%) (Nurhaida & Ichsan, 2018).

$$Packet\ loss = \frac{\gamma}{A} \times 100\%$$

Tabel 2.2 Nilai Packet Loss Standar TIPHON (Mikrotik, n.d.)

Category	Packet Loss (%)
Very Good	0
Good	0-3
Medium	3-15
Bad	15-25

2.4.2 Jitter

Jitter lazimnya disebut variasi *Delay*, berhubungan erat dengan *latency*, yang menunjukkan banyaknya variasi *Delay* pada transmisi data di

jaringan. *Jitter* diakibatkan oleh variasi-variasi dalam panjang antrian, dalam waktu pengolahan data, dan juga dalam waktu penghimpunan ulang paket-paket diakhir perjalanan *jitter* (Nindya *et al,*. 2017). Tujuan pengukuran *jitter* adalah untuk mengetahui kestabilan *forwarding d*ata dalam suatu jaringan. Kestabilan bisa dilihat dari banyaknya variasi *Delay* yang terjadi selama waktu komunikasi di jaringan. Berdasarkan standar TIPHON nilai *jitter* yang sangat bagus adalah 0 s/d 75 ms (Nurhaida & Ichsan, 2018).

$D(0,1)=(R1-R0)-(S1-S0)$

Keterangan:

S1, S0 = Timestamp dari paket 1 dan 0

R1, RO = Waktu Kedatangan Paket 1 dan 0

$$J(1) = J(0) + \frac{(|D1| - J(0))}{16}$$

Tabel 2.3 Nilai *Jitter* Standar TIPHON (Mikrotik, n.d.)

Category	Jitter (ms)
Very Good	0
Good	0 -75
Medium	75-125
Bad	125-225

2.4.3 Throughput

Throughput adalah jumlah total kedatangan paket yang sukses yang diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut. *Throughput* yaitu kecepatan (*rate*) *transfer* data efektif, yang diukur dalam bps (*bit per second*) (Nindya *et al,*. 2017). Tujuan pengukuran *throughput* adalah untuk mengetahui kehandalan jaringan dalam meneruskan paket pada layanan VoIP yang sudah terdapat *network congestion*.

$$\text{Throughput} = \frac{\left(\frac{\text{Paket data diterima}}{\text{lama pengamatan}}\right)}{\text{Besar } \textit{bandwidth}} \times 100\%$$

Tabel 2.4. Nilai *Througput* Standar TIPHON (Mikrotik, n.d.)

Category	*Througput* (kbps)
Perfect	> 2101
Very Good	1201-2100
Good	701-1200
Medium	339-700
Bad	0-338

2.4.4 Delay

Delay adalah waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* dapat dipengaruhi oleh jarak, media fisik, kongesti atau juga waktu proses yang lama (Nindya *et al*,. 2017). Tujuan pengukuran *Delay* untuk mengetahui berapa lama waktu yang dibutuhkan untuk satu paket sampai dari sumber ke tujuan, pada penilitan ini *Delay* yang di ukur merupakan one-way *Delay* (Nurhaida & Ichsan, 2018).

$$Delay = waktu\ paket\ akhir - waktu\ paket\ awal$$

$$Rata - rata\ Delay = \frac{Total\ Delay}{Total\ Paket\ yang\ Diterima}$$

Tabel 2.5 Nilai *Delay* Standar TIPHON (Mikrotik, n.d.)

Category	*Delay* (ms)
Very Good	< 150
Good	150 - 300
Medium	300 - 450
Bad	> 450

2.5 *Router*

Router adalah sebuah perangkat yang digunakan untuk mengirimkan paket data melalui sebuah jaringan menuju tujuannya melalui proses yang disebut *Routing (*Mikrotik, *n.d.). Router* berfungsi sebagai penghubung antara dua atau lebih jaringan yang berada pada jaringan yang berbeda supaya bisa berhubungan, proses pengambilan paket data pada perangkat jaringan kemudian meneruskan data dari satu jaringan ke jaringan lainnya. Proses *Routing* terjadi pada Lapisan 3 OSI yaitu pada *Layer Network* (Regula, 2018). *Router* menggunakan IP address tujuan untuk mengirimkan paket, dan agar *router* mengetahui rute mana yang harus digunakan untuk meneruskan paket ke alamat tujuan, *router* harus belajar atau bertukar informasi sesama *router* yang saling terhubung untuk mengetahui jalur atau rute yang terbaik (Kusniyati,2017).

2.6 *Forwarding Class (FC)*

Router memiliki kemampuan membagi beberapa *forwarding-class*, cara kerjanya mengatur dan menangani bagaimana mekanisme paket di antrean, antrean dibuat menurut prioritas masing-masing *forwarding-class* sebelum paket diteruskan ke dalam *switch-fabric*. Secara umum *forwarding-class* terbagi menjadi tiga kategori utama yaitu, *High-Priority*, *Assured* dan *Best-Effort*. Dalam kategori *high-priority* terdapat empat kelas yaitu, *network-controller* (nc), *expedited* (ef), *high* 1 (h1), *high* 2 (h2). Kategori *high-priority* selalu menjadi prioritas teratas untuk dapat dilewatkan paket ketika terjadi *network congestion*. Pada kategori *assured* terdapat dua kelas yaitu, *assured* (af) dan *low* 1 (Othman *et al,*. 2012). Kategori *assured* hanya menjamin paket sampai ketujuan jika *bandwith* dalam jaringan masih memungkinkan dan tidak terdapat kelas yang lebih tinggi di atasnya. Pada kategori *best-effort* terdapat dua kelas yaitu, *low* 2 (l2) dan *best-effort* (be). Kelas *best-effort* tidak memiliki jaminan pengiriman, semua paket dalam kelas ini dikirim sesuai kemampuan *bandwith*. Kelas *best-effort* juga menjadi kelas *default* bagi pengiriman paket-paket MPLS (Nurhaida & Ichsan, 2018).

Table 2.6 Tipe *Forwarding Class*

FC-ID	*FC Name*	*FC Designation*	*DiffServ Names*
7	*Network Control*	NC	NC2
6	*High-1*	H1	NC1
5	*Expedited*	EF	EF
4	*High-2*	H2	AF4
3	*Low-1*	L1	AF2
2	*Assured*	AF	AF1
1	*Low-2*	L2	CS1
0	*Best Effort*	BE	BE

2.7 3CX

3CX merupakan aplikasi *server* dan *client* VoIP berbasis *open source* untuk *operating system Windows* dengan mendukung protokol SIP dan IAX memungkinkan 3CX dapat terintegrasi lebih mudah dengan *server* VoIP. 3CX memiliki keunggulan mudah terintegrasi otomatis dengan *server* jika berada dalam satu jaringan. 3CX mendukung *protocol* SIP dan IAX. 3CX memilik GUI yang mudah di aplikasikan dan juga pengaturan yang mudah (Hidayat, 2019).

Gambar 2.2 Logo 3CX

2.8 *Iperf*

Iperf merupakan suatu *tool* yang bisa digunakan untuk mengaliri trafik dan juga bisa digunakan untuk mengukur *bandwidth* dalam sebuah *link network*. Untuk mengaliri trafik dibutuhkan *link point to point,* yang memungkinkan untuk di *install* disisi *server* dan juga di sisi *client*. *Iperf* bisa digunakan untuk TCP dan UDP.