

BAB II

DASAR TEORI

2.1. Kajian Pustaka

Penelitian terkait membandingkan kinerja antara protokol *routing* AODV dan DSDV pada jaringan MANET telah dilakukan sebelumnya oleh Nazarudin Faruq Assidiq [3], dalam penelitian ini protokol *routing* serta parameter pembanding yang digunakan sama dengan penelitian penulis, namun pada penelitian penulis juga membandingkan performansi *metric*, seperti: *packet delivery ratio*, *throughput*, dan *end-to-end delay* sebagai parameter pembanding dan tidak hanya konsumsi energi dan energi yang tersisa. Hasil yang didapatkan dalam penelitian Nazarudin Faruq Assidiq ini adalah dalam skenario penambahan node konsumsi energi pada AODV lebih hemat dari DSDV dengan selisih 0,887 Joules dan pada penambahan luas area konsumsi energi node pada AODV lebih hemat daripada DSDV dengan rasio perbandingan konsumsi energi sebesar 0,1%. Beberapa contoh penelitian yang lain telah dilakukan adalah membandingkan kinerja antara protocol *routing* reaktif AODV terhadap *routing* protokol proaktif OLSR dalam penelitian yang dilakukan oleh Arif Mahmud, Mohammad Mohaiminul Islam, Sazid Ahmed Bappi, dan Afridi Kamal Shakil [4], membandingkan kinerja antara protocol *routing* proaktif DSDV dengan protocol *routing* reaktif DSR yang dilakukan oleh R.Kasiram, G. Rajkumar, J.Ashokan, dan D. Parthiban [5], membandingkan kinerja antar sesama protokol *routing* reaktif AODV dan DSR yang dilakukan oleh Tyas Nurfitriana [6], membandingkan kinerja antar sesama protokol *routing* proaktif OLSR, DSDV, WRP yang dilakukan oleh Sandeep Gupta, Dr. Rahul Malhotra, dan Dr. B.S. Dhaliwal [7], serta membandingkan protokol *routing* proaktif, reaktif, dan *hybrid* DSDV, AODV, DSR & ZRP yang dilakukan oleh Meenakshi, Vinod Kumar Mishra,,Kuber Singh [8].

Berbeda dengan penelitian penulis yang membandingkan protokol *routing* AODV dan DSDV, pada penelitian [4] bertujuan untuk membandingkan kinerja kedua protokol *routing* AODV dan OLSR yang

dievaluasi berdasarkan skenario *cluster* pada platform NS2 untuk varian yang berbeda seperti jumlah paket transmisi, delay, packet lost, throughput dll. Hasil penelitian menunjukkan bahwa kinerja OLSR lebih baik daripada AODV dalam hal generasi paket, paket drop, transfer paket dan rasio pengiriman paket di mana hasil yang jauh lebih baik diprediksi untuk jaringan yang lebih besar. Skenario *broadcast* diterapkan dan hasilnya diharapkan dapat digunakan dalam kasus umum. Faktanya semua faktor yang digunakan sangat dekat dengan skenario dunia nyata dan dengan demikian sedikit perbedaan dalam hasil yang diharapkan saat disematkan ke perangkat.

Penelitian [5] bertujuan untuk membandingkan kinerja kedua protokol *routing* AODV dan OLSR yang dievaluasi berdasarkan pengukuran performansi *metric* seperti *throughput*, *packet drop*, *packet delivery ratio*, dan, *average energy consumption* menggunakan NS-2.35 saat *node-node* nya sedang bergerak serta dalam keadaan diam. Hasil yang didapatkan dari penelitian ini membuktikan bahwa protokol *routing* DSR menghasilkan *throughput* yang lebih baik, *packet delivery ratio* yang lebih tinggi dengan *packet drop* serta penggunaan energi yang lebih sedikit sehingga mengungguli performansi DSDV, baik itu saat sedang bergerak maupun dalam keadaan diam.

Penelitian [6] bertujuan untuk membandingkan konsumsi energi dan energi yang tersisa pada protokol *routing* dalam ruang lingkup reaktif yaitu AODV (*Ad hoc On-Demand Distance Vector*) dan DSR (*Dynamic Source Routing*) dengan skenario penambahan jumlah node dan luas area dari jaringan MANET menggunakan Network Simulator 2.

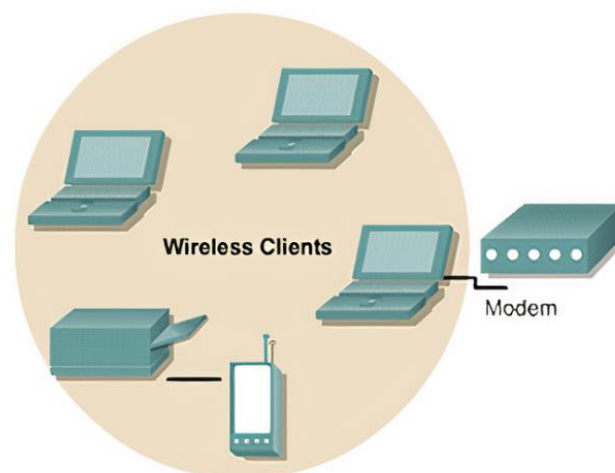
Penelitian [7] bertujuan untuk mengevaluasi kinerja protokol *routing* proaktif melalui MATLAB. Simulasi dilakukan melalui protokol *routing* *Optimized Link State Routing* (OLSR), *Destination-Sequenced Distance-Vector* (DSDV) dan *Wireless Routing Protocol* (WRP) dengan mengevaluasi *Throughput*, *Packet Delivery Ratio (PDR)*, *MAC collision*, dan *Error Rate* untuk protokol tersebut. Hasil evaluasi menunjukkan bahwa OLSR memiliki performa terbaik dibandingkan protokol lainnya.

Penelitian [8] bertujuan untuk melakukan simulasi & perbandingan kinerja antara tiga jenis protokol *routing*, *Table Driven (Proactive)*, *On-Demand (Reactive)* & *Hybrid Protocol* menggunakan alat simulasi NS-2 (2.26). Protokol *routing* ini dibandingkan dalam hal rasio pengiriman paket, delay rata-rata dan kecepatan.

2.2. Jaringan Nirkabel (*Wireless*)

Jaringan nirkabel atau *wireless* adalah sekumpulan perangkat komputer / *client* / *host* yang saling terhubung antara satu dengan yang lainnya dengan menggunakan media transmisi tanpa kabel [9]. Teknologi *wireless* memungkinkan untuk dapat menghubungkan beberapa komputer sekaligus tanpa menggunakan media transmisi fisik seperti kabel, melainkan dengan menggunakan gelombang elektromagnetik dengan frekuensi 2.4 GHz. Jaringan *wireless* memiliki beberapa standarisasi yang sering digunakan antara lain 802.11b, 802.11a, dan 802.11g [10]. Teknologi *nirkabel* mempunyai keunggulan diantaranya biaya pembangunan yang relatif murah, instalasi mudah serta kemampuannya menjangkau area geografis yang lebih luas [11].

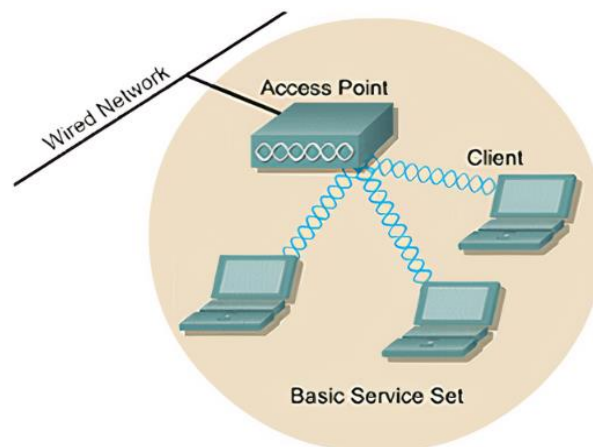
Topologi pada jaringan *wireless* hanya dibagi menjadi dua saja, berbeda dengan jaringan kabel yang mempunyai banyak topologi. Topologi yang digunakan pada jaringan *wireless* menurut komite 802.11 yang bertanggung jawab dalam menangani *Wireless LAN (WLAN)* & *Mesh (Wi-Fi Certification)* adalah topologi *ad hoc* dan topologi *infrastructure* [12].



Gambar 2.1 Topologi Jaringan *Ad-Hoc* [12]

Topologi *ad-hoc* adalah jaringan *wireless* sederhana yang memungkinkan terjadinya komunikasi antara dua komputer atau lebih secara langsung tanpa menggunakan perantara seperti *wireless access point* atau *wireless router*, oleh sebab itu koneksi pada topologi *ad-hoc* juga disebut sebagai koneksi *peer-to-peer* atau *computer-to-computer*.

Beberapa kelemahan dari topologi *ad-hoc* yaitu jangkauan komunikasi antar komputer yang terbatas serta tidak dapat melakukan komunikasi jika dihubungkan dengan komputer yang berkabel. *Independent Basic Service Set (IBSS)* merupakan istilah lain untuk menyebut topologi *ad-hoc* [12].



Gambar 2.2 Topologi Jaringan Infrastructure [12]

Topologi *infrastructure* merupakan kebalikan dari topologi *ad-hoc* dimana membutuhkan adanya perantara yang berupa *wireless access point* (*wireless router*) agar antara dua komputer atau lebih dapat saling berkomunikasi. Pada jaringan *wireless*, *Access point* akan berfungsi sebagai sentral atau pusat jaringan yang dapat menghubungkan beberapa *client* sekaligus, hal ini mirip seperti fungsi *hub* atau *switch* yang digunakan pada jaringan berkabel.

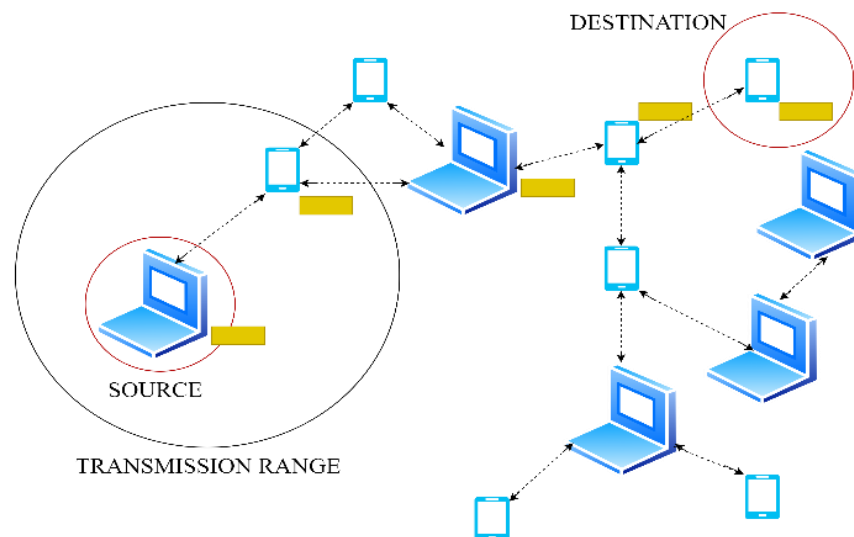
Perangkat *wireless* atau *wireless adapter* dari satu komputer yang terdapat pada topologi *insfrastructure* tidak dapat dihubungkan secara langsung dengan perangkat *wireless* dari komputer lainnya untuk dapat saling berkomunikasi satu sama lain, namun membutuhkan perantara *access point*. *Access point* juga dapat dihubungkan sebagai koneksi dengan jaringan kabel

LAN selain dari fungsi utamanya yang menjadi sentral dalam jaringan. *Basic Service Set* (BSS) merupakan istilah lain untuk menyebut topologi *infrastructure* [12].

2.3. MANET (*Mobile Ad-hoc Network*)

Sebuah MANET memiliki beberapa karakteristik, yaitu [13]:

1. Tidak memiliki infrastruktur terpusat, jaringan *ad hoc* terdesentralisasi dengan semua *node mobile* yang berfungsi sebagai router dan semua perangkat nirkabel saling terhubung satu sama lain.
2. Memiliki topologi dinamis, node bebas bergerak dengan berubah-unah, menyebabkan topologi jaringan berubah dengan cepat dan tidak terduga dari waktu ke waktu.
3. MANET beroperasi pada tautan kapasitas variabel yang dibatasi *bandwidth*, dengan demikian, MANET memiliki tautan *bandwidth* yang relatif rendah, tingkat kesalahan bit yang tinggi, dan tautan yang tidak stabil dan asimetris.
4. MANET sering terikat oleh operasi yang dibatasi energi, Ini karena *node*-nya sering kali merupakan perangkat bertenaga baterai genggam.



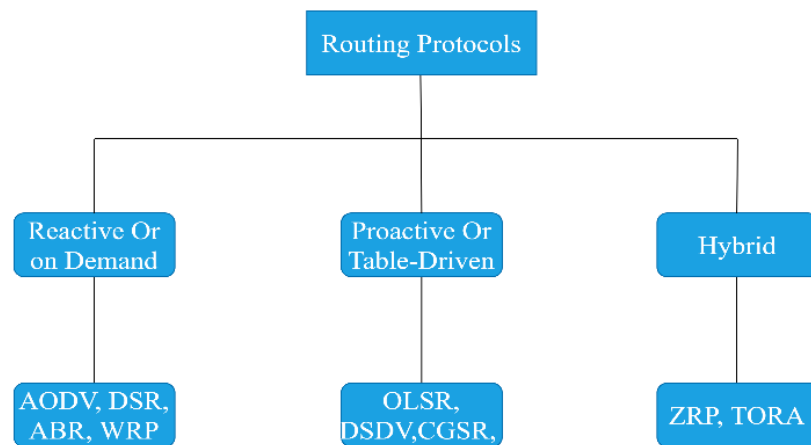
Gambar 2.3 Mobile Ad-Hoc Network [14]

MANET adalah jenis jaringan *ad-hoc* yang dapat berubah lokasi dan mengkonfigurasi sendiri dengan cepat. MANET dapat berupa model koneksi Wi-Fi, atau standar lain, seperti transmisi seluler atau satelit. MANET memiliki banyak kegunaan seperti militer, komunikasi, pertemuan

konferensi, medan perang otomatis, membuat ruang kelas virtual dan dalam jaringan sensor. Fitur utama dari MANET memulihkan dan mengatur diri sendiri dan transmisi melalui *multiple hops*. [14].

2.4. Protokol Routing Pada MANET

Routing merupakan suatu fungsi dari lapisan *network* untuk diimplementasikan pada saat pengiriman paket data dari *node* pengirim menuju ke *node* penerima. *Routing protocol* dapat beradaptasi terhadap perubahan topologi dan trafik pada jaringan MANET yang disebabkan oleh pergerakan *node* yang acak [6]. Klasifikasi protokol *routing* pada MANET, antara lain sebagai berikut:



Gambar 2.4 Klasifikasi Protokol Routing [6]

1. Table-Driven Routing Protocols (Proactive)

Table-Driven Routing Protocols juga disebut sebagai protokol proaktif karena mempertahankan informasi perutean bahkan sebelum memerlukan informasi ini [15] [16]. Setiap node memelihara informasi routing ke setiap node lain dalam jaringan. Informasi rute umumnya disimpan dalam tabel perutean dan diperbarui secara berkala seiring dengan perubahan topologi jaringan. Protokol di dalam kategori ini mempertahankan jumlah tabel yang berbeda. Selain itu tidak cocok untuk jaringan besar, karena perlu memelihara entri untuk setiap *node* dalam tabel *routing*.

2. On-Demand Routing Protocols (Reactive)

Dalam protokol routing Reaktif ketika sebuah node ingin mengirim paket data ke node lain, protokol ini mencari rute sesuai

permintaan dan membuat koneksi untuk mengirim dan menerima paket. Protokol perutean sesuai permintaan biasanya mencakup dua komponen[16]:

- a. *Route Discovery*, jika sumber tidak memiliki rute menuju tujuan dalam tabel *routing* saat ini, maka akan menyiarkan paket penemuan rute ke seluruh jaringan. Setelah rute antara sumber dan tujuan telah ditetapkan, data dapat ditransmisikan melalui rute yang dipilih.
- b. *Route Maintenance*, karena sifat dinamis dari MANET, kegagalan *link* melalui rute dapat terjadi. Pemeliharaan rute adalah mekanisme untuk menangani kerusakan rute. Sebuah *node* dapat mengkonfirmasi jika sebuah paket diterima dengan benar oleh *node* hilirnya (*backward route*) dengan menggunakan salah satu dari tiga jenis pengakuan: *link-level*, pasif (mendengarkan penerusan oleh *node* hop berikutnya), dan lapisan jaringan.

3. *Hybrid Routing Protocols*

Protokol *routing Hybrid* menggabungkan Reaktif dan Proaktif [17]. Protokol ini memanfaatkan keunggulan dari kedua protokol tersebut. Pertama, dapat menggunakan mekanisme Reactive dengan menormalkan delay yang dihasilkan yang dibuat saat Route Discovery dijalankan, dan juga dengan mengurangi Routing Overhead [18]. Kedua, mekanisme yang diterapkan oleh *Hybrid* adalah *Proactive* jika jarak antar node relatif tidak jauh. Di sisi lain, jika jarak antar node sangat tidak moderat, maka Reaktif akan dipilih untuk dieksekusi [17].

2.4.1. AODV

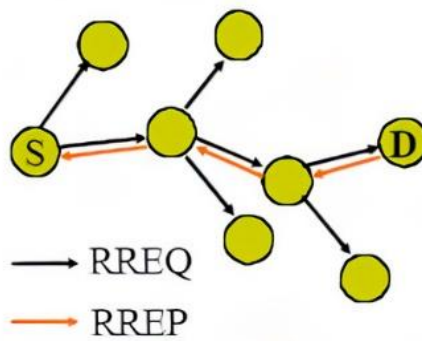
Ad hoc On-demand Distance Vector (AODV) adalah *routing protocol* yang termasuk dalam klasifikasi reaktif *routing protocol*, yang hanya me-request sebuah rute saat dibutuhkan. AODV yang standar ini dikembangkan oleh C. E. Perkins, E.M. Belding-Royer dan S. Das pada RFC 3561 [19].

Ciri utama dari AODV adalah menjaga *timer-based state* pada setiap node sesuai dengan penggunaan *table routing*. Tabel *routing* akan kadaluarsa jika jarang digunakan [20].

AODV memiliki *route discovery* dan *route maintenance*. *Route Discovery* berupa *Route Request* (RREQ) dan *Route Reply* (RREP). Sedangkan *Route Maintenance* berupa *Data*, *Route update* dan *Route Error* (RRER) [20].

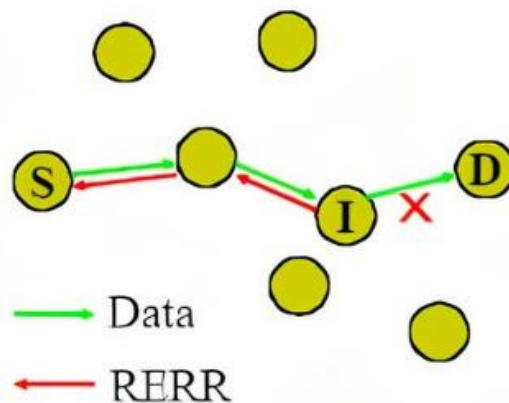
AODV mengadopsi mekanisme yang sangat berbeda untuk menjaga informasi *routing*. AODV menggunakan tabel *routing* dengan satu *entry* untuk setiap tujuan. Tanpa menggunakan *routing* sumber, AODV mempercayakan pada tabel *routing* untuk menyebarkan *RouteReply* (RREP) kembali ke sumber dan secara sekuensial akan mengarahkan paket data menuju ketujuan. AODV juga menggunakan *sequence number* untuk menjaga setiap tujuan agar didapat informasi *routing* yang terbaru dan untuk menghindari *routing loops*. Semua paket yang diarahkan membawa *sequence number* ini [20].

Penemuan jalur (*Path discovery*) atau *Route discovery* diinisiasi dengan menyebarkan *RouteReply* (RREP), seperti terlihat pada Gambar 2.4. Ketika RREP menjelajahi node, ia akan secara otomatis men-*setup path*. Jika sebuah node menerima RREP, maka node tersebut akan mengirimkan RREP lagi ke node atau *destination sequence number*. Pada proses ini, node pertama kali akan mengecek *destination sequence number* pada tabel *routing*, apakah lebih besar dari 1 (satu) pada *RouteRequest* (RREQ), jika benar, maka node akan mengirim RREP. Ketika RREP berjalan kembali ke source melalui path yang telah di-setup, ia akan men-*setup* jalur kedepan dan meng-*update timeout*.



Gambar 2.5 Mekanisme Penemuan Rute [20]

Jika sebuah *link* ke *hop* berikutnya tidak dapat dideteksi dengan metode penemuan rute, maka link tersebut akan diasumsikan putus dan *RouteError* (RERR) akan disebarakan ke node tetangganya seperti terlihat pada Gambar 2.5. Dengan demikian sebuah *node* bisa menghentikan pengiriman data melalui rute ini atau meminta rute baru dengan menyebarkan *RREQ* kembali.

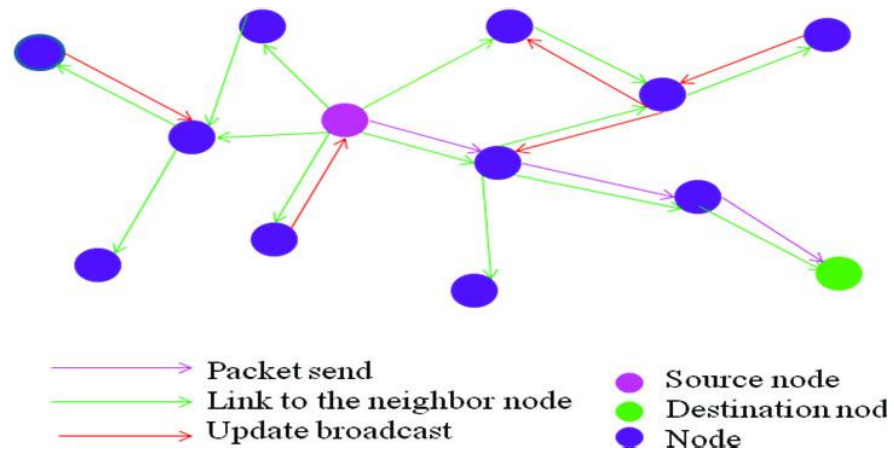


Gambar 2.6 Mekanisme Data dan Route Error [20]

2.4.2. DSDV

Destination Sequenced Distance Vector (DSDV) merupakan *Routing Protocol* routing proaktif atau *table-driven* untuk mengatasi *routing loop*. Pada DSDV, mengirimkan pesan ke jaringan menggunakan *sequenced number*. *Sequenced number* juga digunakan pada saat adanya perubahan dalam jaringan, hal ini terjadi karena DSDV menggunakan *table routing* yang juga merupakan sifat dari *routing* proaktif yang akan selalu memperbarui informasi secara periodik. Pada saat melakukan *update* rute, DSDV menggunakan

time-driven dan event driven. Time-driven merupakan node-node akan saling bertukar informasi dengan node-node lainnya untuk mendapatkan informasi yang terbaru secara periodik. Sedangkan event driven terjadi ketika adanya pembaharuan yang penting, maka node lainnya akan digerakkan oleh trigger atau fenomena tertentu untuk mengirimkan informasi dari routing table yang berubah [21]. DSDV membutuhkan konsumsi energi yang cukup besar karena sifat DSDV yang merupakan *routing* proaktif dimana DSDV harus selalu memperbaharui *routing table* secara periodik. Gambar 2.6 menjelaskan cara kerja DSDV [23].



Gambar 2.7 Cara Kerja DSDV [23]

2.5. QoS (*Quality of Service*)

QoS adalah suatu pengukuran yang dilakukan guna mengetahui performansi kinerja dari suatu jaringan. *Best-effort service*, *integrated service*, serta *differentiated service* merupakan jenis-jenis metode QoS yang sering digunakan [24]

Best-Effort Service adalah metode QoS dengan melakukan semua usaha sehingga suatu paket dapat dikirimkan ke tujuan, oleh karena itu tidak ada jaminan kalau paket tersebut akan diterima oleh yang dimaksud. *Integrated service* melakukan negosiasi parameter jaringan secara *end to end* untuk menyediakan aplikasi dengan tingkat jaminan. Sedangkan *differentiated service* memberikan prioritas tertentu di atas jaringan yang

berbeda pada aplikasi ataupun protokol dengan menyediakan suatu set perangkat klasifikasi dan mekanisme antrian [24].

Delay, jitter, throughput, dan packet loss adalah contoh parameter QoS untuk mengetahui kualitas dari suatu jaringan. *Delay* adalah total waktu yang dibutuhkan dari suatu data yang dikirimkan oleh pengirim dan dapat diterima oleh penerima pada suatu jaringan. *Throughput* adalah total paket yang berhasil tiba ke tujuan selama batas waktu tertentu [24]. Tabel berikut merupakan nilai batasan *delay* yang menunjukkan kualitas layanan berdasarkan ITU-T G.114 ditunjukkan pada table berikut [25]:

Tabel 2.1 Nilai Batasan *Delay* [25]

<i>Delay</i> (ms)	Keterangan	Kategori
0-150	Dapat diterima	Baik
150-400	Dapat diterima, dengan catatan <i>network administrator</i> harus mewaspadai apapun yang bisa mempengaruhi kualitas jaringan.	Cukup
>400	Secara umum tidak dapat diterima, tetapi batas nilai ini dapat berubah untuk kasus-kasus khusus	Buruk

Jitter terjadi akibat adanya selisih waktu kedatangan antar paket di penerima. Semakin besar nilai jitter maka semakin besar pula nilai delay yang dihasilkan [24]. Tabel berikut merupakan nilai batasan *jitter* yang menunjukkan kualitas layanan berdasarkan ITU-T G.114 ditunjukkan pada table berikut [25]:

Tabel 2.2 Nilai Batasan *Jitter* [25]

<i>Delay</i> (ms)	Keterangan	Kategori
0-20	Dapat diterima	Baik
20-50	Dapat diterima	Cukup
>50	Tidak dapat diterima	Buruk

. *Packet loss* merupakan jumlah paket yang hilang saat proses pengiriman terjadi [24]. Tabel berikut merupakan nilai batasan *packet loss* yang menunjukkan kualitas layanan berdasarkan ITU-T G.114 ditunjukkan pada table berikut [25]:

Tabel 2.3 Nilai Batasan *Packet Loss* [25]

<i>Delay</i> (ms)	Keterangan	Kategori
0-1	Dapat diterima	Baik
1-5	Dapat diterima	Cukup
>10	Tidak dapat diterima	Buruk