

**ANALISIS PENGARUH *FRAME RATE* DAN *BIT RATE* VIDEO  
TERHADAP KINERJA VIDEO *STREAMING* PADA JARINGAN  
WLAN 802.11n**

**SKRIPSI**

**TEKNIK ELEKTRO KONSENTRASI TEKNIK TELEKOMUNIKASI**

Diajukan Untuk Memenuhi Persyaratan  
Memperoleh Gelar Sarjana Teknik



**FATHUR RAHMAN**

**NIM. 125060300111047**

**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI**

**UNIVERSITAS BRAWIJAYA**

**FAKULTAS TEKNIK**

**MALANG**

**2016**

## LEMBAR PERSETUJUAN

### ANALISIS PENGARUH *FRAME RATE* DAN *BIT RATE* VIDEO TERHADAP KINERJA VIDEO *STREAMING* PADA JARINGAN WLAN 802.11n

#### SKRIPSI

Ditujukan untuk memenuhi persyaratan  
memperoleh gelar Sarjana Teknik



Disusun oleh:

**FATHUR RAHMAN**

**NIM.125060300111047**

Telah diperiksa dan disetujui oleh:

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Ali Mustofa, S.T., M.T.**  
**NIP. 19710601 20003 1 001**

**Gaguk Asmungi, S.T., M.T.**  
**NIP. 19670627 199802 1 001**

## PENGANTAR

Alhamdulillah, segenap puji dan syukur penulis panjatkan kepada Allah SWT yang telah memberikan rahmat dan hidayahNya sehingga penulis dapat menyelesaikan skripsi dengan judul “Analisis Pengaruh *Frame Rate* dan *Bit Rate* Video Terhadap Kinerja Video Streaming Pada Jaringan Wlan 802.11n”, yang diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Teknik.

Penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada berbagai pihak yang telah membantu dan mendukung dalam penyelesaian skripsi ini, yaitu :

1. Keluarga tercinta terima kasih atas semua kesabaran, dukungan, semangat, pelajaran hidup yang tak ternilai dan segalanya yang telah diberikan dari saya lahir sampai saat ini.
2. Bapak M. Azis Muslim, S.T., M.T., Ph.D selaku Ketua Jurusan Teknik Elektro.
3. Bapak Ali Mustofa, S.T., M.T. dan Gaguk Asmungi, S.T., M.T. selaku dosen pembimbing yang telah sabar membimbing dan memberikan masukan dalam penyelesaian skripsi ini.
4. Ibu Rusmi Ambarwati, S.T., M.T. selaku Ketua Kelompok Dosen Keahlian Teknik Telekomunikasi.
5. Kepada rekan rekan elektro sehat yang sering mengajak diskusi sambil meminum kopi.
6. Terima kasih kepada habib, gezadio dan naufal yang telah sangat membantu dalam penyelesaian skripsi ini.
7. Kepada teman kontrakan menuju sukses terima kasih atas canda dan tawanya, jangan lupa martabak lah.
8. Keluarga kedua, teman-teman angkatan 2012 “Voltage’12” yang tidak bisa saya sebutkan satu persatu. Terima kasih telah menjadi bagian dari perjalanan hidup yang terpenting dan tak terlupakan di kampus.
9. Rekan-rekan Pejuang Paket C 2012 yang tidak bisa disebutkan satu per satu.
10. Semua pihak yang telah membantu penyusunan skripsi ini yang tidak dapat disebut satu per satu.

Penulis menyadari bahwa skripsi ini masih memiliki kekurangan sehingga saran dan kritik membangun sangat penulis harapkan untuk mengisi kekurangan tersebut. Semoga

skripsi ini bermanfaat untuk kita semua, khususnya Jurusan Teknik Elektro Fakultas Teknik  
Universitas Brawijaya.

Malang, 29 Juli 2016

Penulis



## DAFTAR ISI

|                                                      |       |
|------------------------------------------------------|-------|
| <b>PENGANTAR</b> .....                               | i     |
| <b>DAFTAR ISI</b> .....                              | iii   |
| <b>DAFTAR GAMBAR</b> .....                           | vi    |
| <b>DAFTAR TABEL</b> .....                            | viii  |
| <b>DAFTAR LAMPIRAN</b> .....                         | ix    |
| <b>ABSTRAK</b> .....                                 | x     |
| <br><b>BAB I PENDAHULUAN</b> .....                   | <br>1 |
| 1.1. Latar Belakang.....                             | 1     |
| 1.2. Rumusan Masalah .....                           | 2     |
| 1.3. Batasan Masalah.....                            | 2     |
| 1.4. Tujuan.....                                     | 3     |
| 1.5. Manfaat.....                                    | 3     |
| 1.6. Sistematika Penulisan.....                      | 3     |
| <br><b>BAB II TINJAUAN PUSTAKA</b> .....             | <br>5 |
| 2.1. Teknologi <i>Streaming</i> .....                | 5     |
| 2.1.1    Arsitektur Streaming .....                  | 5     |
| 2.1.2    Parameter Kualitas Video Streaming .....    | 6     |
| 2.1.3    Standard Kompresi Streaming.....            | 7     |
| 2.1.3.1    H263 .....                                | 8     |
| 2.1.3.2    MPEG-4.....                               | 8     |
| 2.1.4    Standard Kompresi Audio .....               | 9     |
| 2.1.4.1    G723 .....                                | 9     |
| 2.1.4.2    G729 .....                                | 9     |
| 2.1.5    Protokol Video Streaming .....              | 9     |
| 2.1.5.1    UDP (User Datagram Protokol) .....        | 10    |
| 2.1.5.2    Real-Time Transport Protocols (RTP) ..... | 10    |
| 2.1.5.3    Real-Time Streaming Protocol (RTSP).....  | 11    |
| 2.1.5.4    Real-Time Control Protocol (RTCP).....    | 11    |
| 2.1.6    Live Video Streaming.....                   | 12    |

|                                                 |           |
|-------------------------------------------------|-----------|
| 2.2. Wireless Local Area Network (WLAN) .....   | 12        |
| 2.2.1 Spesifikasi WLAN 802.11 .....             | 12        |
| 2.2.1.1 802.11a .....                           | 12        |
| 2.2.1.2 802.11b .....                           | 13        |
| 2.2.1.3 802.11g .....                           | 13        |
| 2.2.1.4 802.11n .....                           | 13        |
| 2.2.2 Prinsip Kerja WLAN .....                  | 14        |
| 2.3. Quality of Service Video Streaming .....   | 15        |
| 2.3.1 Delay End-to-End .....                    | 15        |
| 2.3.1.1 Delay end- to-end pada Codec .....      | 16        |
| 2.3.1.2 Delay end-to-end jaringan .....         | 17        |
| 2.3.1.2.1 Delay Proses .....                    | 17        |
| 2.3.1.2.2 Delay Transmisi .....                 | 17        |
| 2.3.1.2.3 Delay Propagasi .....                 | 18        |
| 2.3.1.2.4 Delay Antrian .....                   | 18        |
| 2.3.2 Packet Loss .....                         | 18        |
| 2.3.3 Throughput .....                          | 20        |
| 2.4. Perangkat Lunak (Software) Pendukung ..... | 20        |
| 2.4.1. Wireshark .....                          | 20        |
| 2.4.2. Video LAN Client (VLC) .....             | 24        |
| <b>BAB III METODE PENELITIAN .....</b>          | <b>25</b> |
| 3.1. Konfigurasi Jaringan .....                 | 25        |
| 3.2. Pengambilan Data .....                     | 26        |
| 3.2.1. Pengambilan Data Sekunder .....          | 26        |
| 3.2.2. Pengambilan Data Primer .....            | 27        |
| 3.3. Analisis Data .....                        | 29        |
| 3.4. Pengolahan Data Primer .....               | 30        |
| 3.4.1 Delay End-to-End .....                    | 30        |
| 3.4.2 Packet Loss .....                         | 31        |
| 3.4.3 Throughput .....                          | 32        |
| 3.5. Pembahasan dan Hasil .....                 | 33        |
| 3.6. Penarikan Kesimpulan dan Saran .....       | 34        |
| <b>BAB IV HASIL DAN PEMBAHASAN .....</b>        | <b>35</b> |

|                                                                       |    |
|-----------------------------------------------------------------------|----|
| 4.1. Umum.....                                                        | 35 |
| 4.2. Perancangan dan Instalasi .....                                  | 35 |
| 4.3. Langkah-Langkah Proses <i>Streaming</i> pada <i>Server</i> ..... | 37 |
| 4.4. Langkah-Langkah Proses <i>Streaming</i> pada <i>Client</i> ..... | 37 |
| 4.5. Proses Capturing Data Menggunakan Wireshark .....                | 41 |
| 4.6. Hasil dan Pembahasan.....                                        | 43 |
| 4.6.1 Hasil Melalui Eksperimen .....                                  | 44 |
| 4.6.1.1 <i>Delay End-to-End</i> .....                                 | 45 |
| 4.6.1.2 <i>Packet Loss</i> .....                                      | 47 |
| 4.6.1.3 <i>Throughput</i> .....                                       | 49 |
| 4.6.2 Hasil Perhitungan Teoritis.....                                 | 51 |
| 4.6.2.1 Spesifikasi Paket Video <i>Streaming</i> .....                | 51 |
| 4.6.2.2 <i>Delay End-to-End</i> .....                                 | 51 |
| 4.6.2.2.1 <i>Delay CODEC</i> .....                                    | 51 |
| 4.6.2.2.2 <i>Delay Proses</i> .....                                   | 52 |
| 4.6.2.2.3 <i>Delay Transmisi</i> .....                                | 54 |
| 4.6.2.2.4 <i>Delay Antrian</i> .....                                  | 55 |
| 4.6.2.2.5 <i>Delay Propagasi</i> .....                                | 57 |
| 4.6.2.2.6 <i>Delay end-to-end total</i> .....                         | 57 |
| 4.6.2.3 <i>Packet Loss</i> .....                                      | 59 |
| 4.6.2.4 <i>Throughput</i> .....                                       | 62 |
| 4.7. Analisis dan Pembahasan Data.....                                | 64 |
| <b>BAB V PENUTUP</b> .....                                            | 66 |
| 5.1 Kesimpulan.....                                                   | 66 |
| 5.2 Saran.....                                                        | 66 |
| DAFTAR PUSTAKA.....                                                   | 67 |
| LAMPIRAN .....                                                        | 68 |

## DAFTAR GAMBAR

|                                                                                                            |    |
|------------------------------------------------------------------------------------------------------------|----|
| Gambar 2.1 Proses Gambar Bergerak .....                                                                    | 7  |
| Gambar 2.2 Prinsip Kerja WLAN .....                                                                        | 15 |
| Gambar 2.3 Contoh Tampilan Wireshark yang Sedang Meng-capture Paket-Paket Pada Suatu Sistem Jaringan ..... | 21 |
| Gambar 2.4 Tampilan awal VLC media player .....                                                            | 22 |
| Gambar 3.1 Konfigurasi Jaringan .....                                                                      | 23 |
| Gambar 3.2 Diagram Alir Perhitungan <i>Delay end-to-end</i> .....                                          | 25 |
| Gambar 3.3 Diagram Alir Perhitungan <i>Packet Loss</i> .....                                               | 25 |
| Gambar 3.4 Diagram Alir Perhitungan <i>Throughput</i> .....                                                | 25 |
| Gambar 3.5 Diagram Alir Pengambilan Data Primer .....                                                      | 26 |
| Gambar 3.6 Proses Pengambilan Data Menggunakan Wireshark.....                                              | 27 |
| Gambar 3.7 Diagram alir pengolahan data <i>Delay End-to-End</i> .....                                      | 29 |
| Gambar 3.8 Diagram alir pengolahan data <i>Packet Loss</i> .....                                           | 30 |
| Gambar 3.9 Diagram alir pengolahan data <i>Throughput</i> .....                                            | 31 |
| Gambar 3.10 Pembahasan dan Hasil.....                                                                      | 32 |
| Gambar 4.1 Wireless Router TP-LINK TL-WR74IND.....                                                         | 34 |
| Gambar 4.2 Koneksi ke Wireless Router pada jaringan WLAN .....                                             | 35 |
| Gambar 4.3 PING menuju Destinasi.....                                                                      | 36 |
| Gambar 4.4 Tampilan menu Media VLC.....                                                                    | 36 |
| Gambar 4.5 Kotak dialog pemilihan video .....                                                              | 36 |
| Gambar 4.6 Kotak dialog destination setup .....                                                            | 37 |
| Gambar 4.7 Kotak dialog transcoding options.....                                                           | 37 |
| Gambar 4.8 Kotak Dialog stream output .....                                                                | 37 |
| Gambar 4.9 Tampilan menu media VLC .....                                                                   | 38 |
| Gambar 4.10 Kotak dialog pengisian alamat video <i>streaming</i> .....                                     | 38 |
| Gambar 4.11 Tampilan video <i>streaming</i> .....                                                          | 39 |
| Gambar 4.12 Tampilan <i>Interface List</i> Wireshark.....                                                  | 39 |
| Gambar 4.13 Tampilan <i>Capture</i> menu.....                                                              | 40 |
| Gambar 4.14 Tampilan pada <i>Capture Interface</i> .....                                                   | 40 |

|                                                                                                                                                      |    |
|------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Gambar 4.15 Tampilan <i>Capturing</i> data Wireshark .....                                                                                           | 40 |
| Gambar 4.16 Grafik hubungan antara <i>Bit rate</i> dan <i>Frame rate</i> video dengan <i>Delay end-to-end</i> hasil <i>capturing</i> wireshark ..... | 42 |
| Gambar 4.17 Grafik hubungan antara <i>Bit rate</i> dan <i>Frame rate</i> video dengan <i>packet loss</i> hasil <i>capturing</i> wireshark.....       | 44 |
| Gambar 4.18 Grafik hubungan antara <i>Frame Rate</i> dan <i>Bit Rate</i> video dengan <i>throughput</i> hasil <i>capturing</i> wireshark.....        | 45 |
| Gambar 4.19 Proses Delay Enkapsulasi dan Dekapsulasi.....                                                                                            | 47 |
| Gambar 4.20 Grafik hubungan antara <i>Bit rate</i> dan <i>Frame Rate</i> video dengan <i>delay end-to-end</i> secara teoritis.....                   | 54 |
| Gambar 4.21 Grafik hubungan antara <i>Bit rate</i> dan <i>Frame rate</i> video dengan <i>Packet loss</i> secara teoritis.....                        | 56 |
| Gambar 4.22 Grafik hubungan antara <i>Bit rate</i> dan <i>Frame rate</i> video dengan <i>Throughput</i> secara teoritis.....                         | 58 |



## DAFTAR TABEL

|                                                                                |    |
|--------------------------------------------------------------------------------|----|
| Tabel 2.1 Fungsi kerja layer pada video <i>streaming</i> .....                 | 10 |
| Tabel 2.2 Spesifikasi 802.11 .....                                             | 12 |
| Tabel 2.3 Pengelompokan <i>Delay</i> Berdasarkan ITU-T G.114 .....             | 16 |
| Tabel 2.4 Standar tingkat paket hilang berdasarkan ITU-T-G1010 .....           | 19 |
| Tabel 4.1 Fungsi Perangkat Keras .....                                         | 33 |
| Tabel 4.2 Spesifikasi TP-LINK TL-WR74IND .....                                 | 34 |
| Tabel 4.3 Spesifikasi <i>PC Server</i> .....                                   | 34 |
| Tabel 4.4 Tipe Laptop <i>Client</i> .....                                      | 35 |
| Tabel 4.5 Nilai <i>delay end-to-end</i> hasil <i>capturing</i> wireshark ..... | 42 |
| Tabel 4.6 Nilai <i>Packet Loss</i> hasil <i>capturing</i> wireshark .....      | 43 |
| Tabel 4.7 Nilai <i>Throughput</i> hasil <i>capturing</i> wireshark .....       | 45 |
| Tabel 4.8 Nilai <i>Delay</i> Proses .....                                      | 49 |
| Tabel 4.9 Nilai <i>Delay</i> Transmisi .....                                   | 50 |
| Tabel 4.10 Nilai <i>Delay</i> Antrian .....                                    | 52 |
| Tabel 4.11 Nilai <i>Delay end-to-end</i> .....                                 | 53 |
| Tabel 4.12 Nilai <i>Packet loss</i> .....                                      | 56 |
| Tabel 4.13 Nilai <i>Throughput</i> .....                                       | 58 |

## DAFTAR LAMPIRAN

|                                                                                 |    |
|---------------------------------------------------------------------------------|----|
| Lampiran 1 Spesifikasi <i>Wireless Router</i> .....                             | 68 |
| Lampiran 2 Spesifikasi PC Server .....                                          | 69 |
| Lampiran 3 Spesifikasi PC Client .....                                          | 70 |
| Lampiran 4 Pengaruh jumlah client dengan performansi jaringan WLAN 802.11n..... | 74 |
| Lampiran 5 Pengaruh jarak dengan performansi jaringan WLAN 802.11n.....         | 75 |



## ABSTRAKSI

**Fathur Rahman**, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya, Juli 2016, Analisis Pengaruh *Frame Rate* dan *Bit Rate* Video Terhadap Kinerja Video Streaming Pada Jaringan Wlan 802.11n, Dosen Pembimbing : Ali Mustofa, S.T., M.T. dan Gaguk Asmungi, S.T., M.T.

*Video streaming* merupakan teknologi telekomunikasi yang bersifat *real time* serta dapat menyalurkan informasi berupa audio dan video. *Frame rate* dan *bit rate* adalah salah satu faktor yang mempengaruhi kualitas video streaming. *Frame rate* video adalah jumlah bingkai gambar atau frame yang ditunjukkan setiap detik dalam membuat gambar bergerak, diwujudkan dalam satuan fps (*frames per second*). *Bit rate* merupakan jumlah informasi, atau detail yang disimpan per unit waktu dalam suatu rekaman file, terlebih pada video dan audio.

*Wireless Local Area Network* yaitu suatu jenis jaringan komputer yang menggunakan gelombang radio sebagai alat atau media transmisi data. Standar 802.11n adalah sebuah standar jaringan nirkabel yang bekerja pada frekuensi 2,45 GHz dan mampu mencapai kecepatan 300 Mbps. Dalam skripsi ini, akan diteliti pengaruh *frame rate* dan *bit rate* video streaming terhadap kinerja video streaming pada jaringan WLAN 802.11n.

Parameter yang digunakan untuk menentukan *Quality of Service* (QoS) adalah *delay end to end*, *packet loss*, dan *throughput* yang dihitung dengan pendekatan perhitungan teoritis dan pengamatan langsung menggunakan perangkat analisis jaringan (Wireshark). Kualitas performansi layanan *live streaming* pada jaringan *Wireless Local Area Network* 802.11n (WLAN) adalah sesuai dengan standar ITU-T G.1010 dengan nilai *packet loss* <5% untuk nilai *frame rate* 10 fps sampai 40 fps dengan semua variasi *bit rate*, dan nilai *frame rate* 60 fps untuk *bit rate* 256 Kbps dan 512 Kbps layanan dikategorikan baik, Sementara nilai *packet loss* untuk nilai *frame rate* 60 fps dan *bit rate* 1024 dikategorikan cukup karena nilai *packet loss* 5-10%. Nilai *delay end-to-end* untuk semua variasi *bit rate* dan *frame rate* video <150 ms. Dengan nilai <150 ms maka jaringan memiliki kualitas yang baik menurut ITU-G.144.

Kata Kunci – *Frame rate*, *Bit Rate*, QoS, WLAN 802.11n

## BAB I PENDAHULUAN

### 1.1 Latar Belakang

*Streaming* merupakan sebuah teknologi yang mampu mengompresi atau menyusutkan ukuran file audio dan video agar mudah ditransfer melalui jaringan internet. Video *streaming* merupakan teknologi telekomunikasi yang bersifat *real time* serta dapat menyalurkan informasi berupa audio dan video. Dengan teknologi ini *client* tak perlu menunggu hingga file selesai di-*download* secara keseluruhan untuk memainkannya. Terkadang kualitas video dari video *streaming* yang dihasilkan terkadang jelek. Kualitas video *streaming* dipengaruhi oleh beberapa faktor yaitu *Frame size*, *Bit rate*, *Frame rate* dan *Sample rate*. Dalam hal ini penulis lebih menekankan pengujian terhadap *Frame rate* Video dan *Bit rate*. *Frame rate* video adalah jumlah bingkai gambar atau *frame* yang ditunjukkan setiap detik dalam membuat gambar bergerak, diwujudkan dalam satuan fps (*frames per second*), semakin tinggi angka fps maka semakin halus gambar yang dihasilkan atau digerakkan. Sementara itu *bit rate* merupakan jumlah informasi, atau detail yang disimpan per unit waktu dalam suatu rekaman file, terlebih pada video dan audio. Semakin tinggi *bit-rate* gambar menjadi semakin baik kedalaman warnanya.

Terdapat permasalahan yang mempengaruhi video *streaming* yaitu ketika video berhenti ketika saat sedang menonton video dan *client* harus menunggu hingga video tersebut berjalan kembali. Agar video *streaming* berjalan dengan baik maka pentransmisian paket-paket data video *streaming* harus berjalan dengan tepat sehingga tidak terjadi gangguan dan kualitas video yang dihasilkan dapat memuaskan *client*. Oleh karena itu dibutuhkan *Quality of Service* sehingga *client* mendapatkan performansi yang handal dalam video *streaming*. Adapun parameter QoS adalah *delay*, *packet loss ratio*, *throughput*. Berdasarkan ITU-T G.1010 *packet loss ratio* dianggap baik apabila nilainya  $<1\%$ . Dan Berdasarkan ITU-T G.114 kualitas jaringan baik dengan *delay* 0-150 ms. QoS sangat ditentukan oleh kualitas jaringan yang digunakan. Dalam skripsi ini jaringan yang digunakan adalah *Wireless Local Area Network*.

*Wireless Local Area Network* yaitu suatu jenis jaringan komputer yang menggunakan gelombang radio sebagai alat atau media transmisi data. Informasi atau data ditransfer dari satu komputer ke komputer yang lainnya menggunakan gelombang radio. Pada jaringan *Wired Local Area Network*, aplikasi multimedia dapat berjalan dengan baik. Akan tetapi berbeda halnya jika diterapkan pada jaringan yang menggunakan koneksi *Wireless Local Area Network*. Jaringan *Wireless Local Area Network* mempunyai kecepatan pengiriman data yang lebih rendah dibandingkan dengan jaringan *Wired Local Area Network*. Berdasarkan standart *IEEE 802.3ae* kecepatan jaringan *Wired Local Area Network* mencapai 10 Gbps (Forouzan, 2007:412). Sementara berdasarkan *IEEE 802.11n* kecepatan jaringan *Wireless Local Area Network* mencapai 300 Mbps (Forouzan, 2009:434). Dalam skripsi ini, akan diteliti pengaruh *frame rate* dan *bite rate* video *streaming* terhadap kinerja video *streaming* pada jaringan WLAN 802.11n.

### 1.2 Rumusan Masalah

Berdasarkan permasalahan yang ada, maka rumusan masalah dalam penulisan skripsi ini adalah:

1. Bagaimana merancang infrastruktur *live video streaming* pada jaringan WLAN 802.11n ?
2. Bagaimana pengaruh *frame rate* dan *bit rate* video terhadap performansi video *streaming* pada jaringan *Wireless Local Area Network* 802.11n meliputi *packet loss ratio*, *delay end-to-end*, *throughput* terhadap standar ITU-T G.1010 ?

### 1.3 Batasan Masalah

Dalam skripsi ini ruang lingkup pembahasan antara lain :

1. Pengujian dilakukan dengan merubah *frame rate* video mulai dari 10 fps, 25 fps, 40 fps, 60 fps dan *bit rate* video 256 Kbps, 512 Kbps, 1024 Kbps.
2. Menggunakan jaringan WLAN 802.11n yang terdiri atas 1 buah *wireless router* dan 1 buah laptop sebagai *server* dan 5 buah laptop sebagai *client*.
3. Aplikasi yang digunakan sebagai video *live broadcast* adalah VideoLAN *client* pada sisi *server* dan *client*.
4. Parameter QoS antara lain: *packet loss ratio*, *delay*, *throughput*. .

### 1.4 Tujuan

Skripsi ini bertujuan untuk mengetahui dan menganalisi pengaruh *frame rate* dan *bit rate* video *streaming* terhadap performansi jaringan WLAN 802.11n. Serta menganilisis parameter QoS yang meliputi : probabilitas *packet loss ratio*, *delay*, *throughput*. Dengan

hasil analisis nantinya menjadi bahan pertimbangan terhadap kualitas layanan video *streaming* pada jaringan WLAN 802.11n.

### 1.5 Manfaat

Adapun manfaat yang diperoleh dari penulisan skripsi ini, diantaranya sebagai berikut:

1. Dapat digunakan sebagai bahan pembelajaran mengenai teknologi *streaming*, serta pengembangan lebih lanjut peningkatan layanan video *streaming* pada jaringan WLAN.
2. Masyarakat menikmati layanan video *streaming* dengan cakupan layanan yang lebih baik serta dapat meningkatkan kepuasan dan kenyamanan dalam penggunaan layanan.
3. Bagi Penyelenggara Layanan Telekomunikasi, dapat digunakan sebagai bahan acuan dalam meningkatkan performansi layanan video *streaming*, khususnya pada WLAN 802.11n.

### 1.6 Sistematika Penulisan

Sistematika penulisan dan gambaran untuk setiap bab pada tugas akhir ini adalah sebagai berikut :

#### BAB I PENDAHULUAN

Memuat latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan.

#### BAB II TINJAUAN PUSTAKA

Membahas tentang konsep dasar video *streaming* dan *Wireless local area network*, parameter kinerja video *streaming* pada jaringan *Wireless local area network* 802.11n, serta membahas tentang perangkat lunak yang digunakan.

#### BAB III METODE PENELITIAN

Memuat konfigurasi jaringan yang digunakan, metode pengambilan data yang digunakan dalam skripsi, metode analisis data, dan pengambilan kesimpulan.

#### BAB IV PEMBAHASAN DAN HASIL

Menjelaskan langkah untuk membangun konfigurasi jaringan, uji coba dan pengambilan data, kemudian membahas performansi jaringan dilihat dari parameter-parameternya yaitu *throughput*, *delay*, dan *packet loss*

menggunakan *software* WireShark serta melakukan perhitungan secara teoritis.

## BAB V PENUTUP

Berisi kesimpulan dan saran yang diperoleh dari analisis perhitungan yang telah dilakukan serta pemberian saran.



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teknologi Streaming

*Streaming* adalah proses pengiriman media data dari *source* menuju *client* secara *real time*. Proses ini berjalan secara kontinyu dan tidak memerlukan penyimpanan lokal untuk media datanya. Dengan semakin majunya teknologi dapat memudahkan pemilik jasa multimedia untuk mencari cara mengirimkan media data yang berupa file audio video melalui jaringan berbasis Internet Protokol. Komunikasi menggunakan audio dan video lebih interaktif dibandingkan dengan komunikasi lewat huruf dan gambar yang sering digunakan pada awal perkembangan internet.

Saat file video atau audio di *stream*, akan berbentuk sebuah *buffer* di komputer *client*, dan data video - audio tersebut akan mulai di download ke dalam *buffer* yang telah terbentuk pada mesin client. Dalam waktu yang singkat, *buffer* telah terisi penuh dan secara otomatis file video audio dijalankan oleh sistem. Sistem akan membaca informasi dari *buffer* dan tetap melakukan proses download file, sehingga proses *streaming* tetap berlangsung.

##### 2.1.1 Arsitektur Streaming

Media *streaming* terdiri dari streaming video dan streaming audio. Streaming menunjukkan transmisi satu arah dari server ke *client*. Pada sisi client, data masuk ke *buffer* sebelum mulai dikirim ke layar, sehingga terdapat *delay* pada pengiriman data paket. Arsitektur Media Streaming umumnya terdiri dari :

1. *Capture* dan *encoding*
2. Pelayanan oleh server (*Serving*)
3. Distribusi pengiriman
4. *Media Player*

*Capture* dan *encoding* adalah proses pengambilan data audio video dari *microphone* dan kamera serta memprosesnya menjadi sebuah file yang terkompresi. File ini akan disimpan pada server penyimpanan data yang mempunyai *software* khusus untuk bisa mengontrol pengiriman data di-*stream* secara *real time*.

Pada proses *serving*, file yang telah di-encode-kan di kirim ke *server* untuk didistribusikan melalui jaringan. *Server* pada *streaming* yang berbasis web, memiliki dua fungsi pokok, pertama sebagai *web server*, *web server* ini berfungsi untuk mengatur komunikasi antara *client* dengan *server streaming*. Kedua, berfungsi untuk mengontrol pengiriman data *stream* melalui jaringan.

Jaringan distribusi menghubungkan antara *server* dengan video *player*. Proses ini akan melibatkan banyak interkoneksi jaringan dan buffer pada *server*. Koneksi jaringan bisa berasal dari jaringan *local area network* (LAN) menuju jaringan internet.

Video *player* biasanya sebagai tambahan (*plug-in*) pada web browser, yang berfungsi untuk menerima data *stream* dan melakukan dekompresi kembali ke format audio video semula, dimana data tersebut agar bisa dijalankan di komputer (PC) (Austerberry, 2005;140).

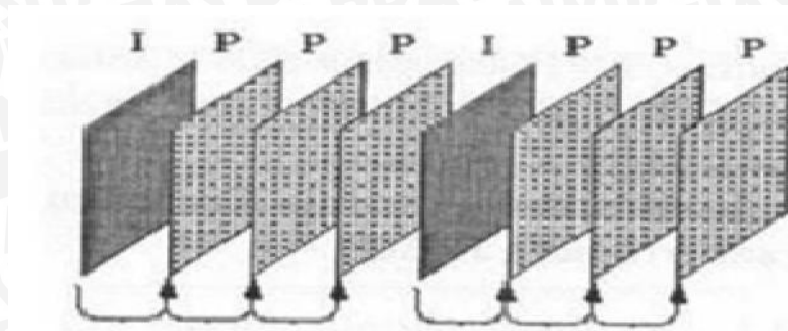
### 2.1.2 Parameter Kualitas Video Streaming

Dalam *streaming* ada beberapa parameter yang dapat mempengaruhi kualitas video *streaming* yaitu *Frame size*, *Bit rate*, *Frame rate* dan *Sample rate*. Keempat parameter tersebut sangat penting pada sebuah *streaming*.

*Frame size* adalah ukuran gambar dalam pixels dan merupakan gabungan dari tinggi dan lebar. Ukuran yang biasa digunakan dalam *streaming* adalah 320x240 (biasa disebut QVGA), 176x144, 169x120. Pemilihan ukuran gambar berhubungan dengan kebutuhan *bandwidth* yang diperlukan.

*Bit rate* merupakan jumlah informasi, atau detail yang disimpan per unit waktu dalam suatu rekaman file, terlebih pada video dan audio. *Bit rate* dapat tergantung pada frekuensi sampel, jenis *encoding*. Semakin tinggi *bit-rate* maka akan semakin banyak informasi data videonya. Oleh karena itu, gambar akan menjadi semakin baik kedalaman warnanya.

*Frame rate* video adalah jumlah bingkai gambar atau *frame* yang ditunjukkan setiap detik dalam membuat gambar bergerak, diwujudkan dalam satuan fps (*frames per second*), semakin tinggi angka fps maka semakin halus gambar yang dihasilkan atau digerakkan, akan tetapi hal ini membuat ukuran file menjadi lebih besar.



Gambar 2.1 Proses gambar Bergerak  
Sumber: Austerberry (2004:368)

*Sample rate* adalah jumlah *sample* per *second* dari file audio yang diukur dalam kilohertz. *Sample rate* yang lebih tinggi akan menghasilkan suara yang lebih baik. Hal ini digambarkan dengan frekuensi yang tinggi tapi akan menyebabkan ukuran file lebih besar. Normalnya, suara diambil *sample* nya pada 44,1 KHz. Ketika *sample* suara tersebut di-*encoding* untuk menaikkan atau menurunkan *rate* nya maka beberapa noise bisa saja terjadi.

*Transcoding* adalah proses pengubahan suatu format kompresi media ke format lainnya dalam sebuah paket data tepat sebelum paket tersebut ditransmisikan. Proses ini berlangsung secara kontinyu selama transmisi berjalan.

### 2.1.3 Standard Kompresi Streaming

Video *codec* standard terpenting untuk melakukan streaming adalah H.261, H.263, MJPEG, MPEG1, MPEG2, dan MPEG4. Dibandingkan dengan video *codec* yang digunakan pada CD-ROM atau siaran TV, *codec* yang didesain untuk jaringan membutuhkan skalabilitas yang lebih tinggi, kompleksitas komputasi yang lebih rendah, toleransi yang lebih besar pada kehilangan data pada jaringan, dan latensi *encode/decode* yang lebih rendah untuk melakukan video *conferencing*. Sebagai tambahan, *codec* diharapkan mampu berkaitan erat dengan aplikasi untuk mencapai kemungkinan terbesar dalam *frame rates* dan kualitas gambar.

Dengan standard tersebut, terlihat jelas bahwa sekarang ini belum ada yang benar-benar ideal untuk video internet. Kenyataannya, dalam beberapa tahun terakhir ini telah berkembang beberapa algoritma baru yang secara spesifik di desain untuk keperluan internet dan semakin lama semakin cocok untuk keperluan internet. Kebanyakan penelitian sedang melirik pada kemampuan skalabilitas, fleksibilitas *codec*, dan pengembangan *codec* yang sudah ada menggunakan *transcoding* dan *filters*. memicu pula pengembangan standard *framework* dari aplikasi seperti H323/H324 untuk

video conference dan MPEG4, yang memudahkan penggabungan inovasi baru dari codec dengan aplikasi video player yang telah dikembangkan, tanpa kerja ulang yang cukup signifikan. Standard Kompresi Video Dibawah ini dijelaskan dua jenis standard kompresi video pada streaming yaitu H263 dan MPEG-4.

### 2.1.3.1 H263

H263 merupakan kompresi video standard dari ITU- T yang didesain untuk komunikasi dengan *bitrate* yang rendah. Diharapkan kemudiannya kompresi video ini dapat digunakan pada *bitrate* dengan kecepatan yang lebih tinggi. Diperkirakan H263 akan menggantikan H261 dalam banyak aplikasi. Penuangan algoritma dari H263 mirip dengan H261, tetapi dengan beberapa perbaikan dan perubahan untuk meningkatkan performa dan penanganan *error*. Perbedaan antara algoritma H261 dengan H263 adalah sebagai berikut. Setengah presisi pixel digunakan pada H263 untuk kompensasi gerakan, yang dimana pada H261 menggunakan presisi pixel penuh.

Beberapa bagian dari struktur data stream telah menjadi pilihan opsional, sehingga codec bisa dikonfigurasi untuk *bit rate* yang lebih rendah atau untuk penanganan *error* yang lebih baik. Ada empat pilihan opsional yang tersedia untuk meningkatkan performa: 8rotoc gerakan yang tidak dibatasi, *coding* aritmatik berbasis sintaks, perkiraan yang lebih baik, dan prediksi *frame* kedepan dan kebelakang mirip dengan format MPEG yang disebut P-B *frames*. H263 mendukung 5 resolusi, yaitu SQCIF, 4CIF, dan 16CIF, sebagai tambahan pada QCIF dan CIF yang telah didukung pada H261. SQCIF kira-kira bemiJai setegah resolusi dari QCIF. 4CIF dan 16CIF bemiJai 16 kali lebih besar dari resolusi CIF. Dengan adanya dukungan dari resolusi 4CIF dan 16CIF membuat codec ini mampu bersaing dengan standard codec video yang memiliki *bitrate* lebih besar seperti MPEG (Salkintzis, 2005:42).

### 2.1.3.2 MPEG-4

Maksud dari pengembangan MPEG-4 adalah memberikan sebuah skema kompresi yang cocok untuk video conference, sekitar kurang dari 64 Kbit per detik. MPEG-4 didasari dari segmentasi dari adegan audiovisual menjadi AVOs atau “AudioNideo Objects”, dimana dapat dilipatgandakan untuk transmisi melewati jaringan heterogen. Sekarang ini pengembangan MPEG-4 *framework* difokuskan pada sebuah 8rotoc yang disebut MSDL (*MPEG-4 Syntatic Description Language*). MSDL memungkinkan aplikasi membangun codec baru dengan cara menyusun beberapa komponen protokol-protokol dan memberikan kemampuan *men-download* komponen-komponen tersebut secara dinamis melalui internet. Filosofi ini foto dengan yang digunakan pada API

multimedia yang dikembangkan oleh *Java Sun Microsystems*, dimana yang memungkinkan untuk secara dinamis men-download komponen *codec*. Trend ini juga dapat dilihat pada produk dari produsen TI terkenal seperti Microsoft dan Netscape, dimana yang mengizinkan untuk beberapa *codec* audio dan video digabungkan dalam solusi *real-time streaming* (Salkintzis, 2005:44).

#### 2.1.4 Standard Kompresi Audio

Kompresi audio atau dikenal juga sebagai pengkodean perseptual (*perceptual coding*) memanfaatkan keterbatasan *9rotoc* pendengaran manusia yang tidak bisa mendengarkan dua buah sinyal dengan frekuensi berdekatan. Dibawah ini dijelaskan dua standard kompresi audio pada *streaming* yaitu, G723 dan G729.

##### 2.1.4.1 G723

G723 adalah kompresi audio standard yang direkomendasikan oleh ITV- T untuk pentransmisian media audio pada kecepatan 5.3 atau 6.3 kbit/s. Pada G723, Audio analog dipotong-potong menjadi potongan sebesar 30 milidetik dan tiap potongan di-encode menjadi 20 atau 24 byte data digital. Pada aplikasi *streaming*, digunakan paket sebesar 24 byte karena paket pada ukuran ini cenderung lebih dapat bertahan ketika ditransmisikan melalui sebuah jaringan. Format ini mendukung kemampuan VAD (*Voice Activity Detection*), yang dapat menyisipkan SID (*Silence Insertion Descriptor*) ketika tidak ada aktifitas audio, sehingga dapat memperkecil ukuran data (Austerberry, 2005:124).

##### 2.1.4.2 G729

ITV- T merekomendasikan kompresi audio standard G729 untuk pentransmisian media audio pada kecepatan 8 kbit/s menggunakan algoritma CS-ACELP (*Conjugate-Structure Algebraic-Code-Excited-Linear Predication*). Pada format ini, audio analog akan dipotong-potong menjadi potongan sebesar 20 milidetik, lalu tiap potongan akan di-encode menjadi data berukuran 20 bytes.

#### 2.1.5 Protokol Video Streaming

Protokol adalah sebuah peraturan yang ditetapkan dan diterapkan suatu teknologi tertentu. Pada teknologi *streaming*, terdapat *9rotocol* yang memang khusus diciptakan untuk mengeksekusi proses *streaming* berupa konten video/audio. Pada teknologi *streaming*, layer-layer mode OSI yang berperan sebagai penunjang antara dua perangkat ataupun banyak perangkat terdapat pada *Application Layer*, *Presentation Layer*, *Session layer* dan *Transport Layer*. Penjelasan secara ringkas tersebut terdapat tabel 2.1.

Tabel 2.1 Fungsi kerja layer pada video streaming

| Layer                     | Fungsi                                                                       |
|---------------------------|------------------------------------------------------------------------------|
| <i>Application Layer</i>  | Antarmuka user berkomunikasi pada jaringan <i>stream</i>                     |
| <i>Presentation Layer</i> | Menerjemahkan konten video/audio kedalam format yang dapat dibaca oleh mesin |
| <i>Session layer</i>      | Mengatur koneksi <i>streaming</i> baik Saat mulai maupun akhir               |
| <i>Transport Layer</i>    | Membawa data <i>stream</i> dari satu <i>device</i> ke <i>device</i> lain     |

Sumber : Salkintzis A (2005:146)

Biasanya koneksi pada jaringan internet menggunakan TCP (*Transport Control Protokol*) sebagai protokolnya. Namun hal ini secara umum tidak berlaku kepada teknologi *streaming*. Protokol yang lazim digunakan pada teknologi streaming adalah UDP (*User Datagram Protokol*). UDP tidak membutuhkan proses *acknowledgment* saat komunikasi antar device dibangun. Karenanya UDP mampu mentransfer data lebih cepat dibanding TCP. Pada *streaming* berupa video/audio, protokol yang dipakai adalah RTP (*Real-Time Transport Protokol*). RTP bekerja di atas UDP. RTP memiliki fungsi untuk mentransmisikan bit-bit video/audio.

#### 2.1.5.1 UDP (*User Datagram Protokol*)

*User Datagram Protokol* merupakan protokol yang bersifat *connectionless*. UDP memungkinkan sebuah aplikasi mengirimkan datagram tanpa perlu menciptakan koneksi terlebih dahulu antara *client* dan *server*. UDP datagram terdiri atas *header* dan *payload*, besar *header* UDP adalah 8 byte. Header UDP terdiri atas port asal, port tujuan, panjang UDP dan checksum UDP tidak melakukan *flow control*, *error control* ataupun melakukan retransmisi (pengiriman ulang UDP datagram). UDP sangat cocok untuk aplikasi *client server*. *Client* terkadang hanya ingin mengirimkan permintaan singkat dan mengharapkan balasan yang segera. Pengkodean yang lebih mudah, pengiriman paket yang lebih sedikit, dan tidak diperlukannya inisialisasi awal koneksi membuat UDP banyak digunakan oleh aplikasi *real-time*.

#### 2.1.5.2 *Real-Time Transport Protocols* (RTP)

Protokol RTP menyediakan transfer media secara *real-time* pada paket jaringan. Protokol RTP menggunakan UDP dan header RTP mengandung informasi kode bit yang spesifik pada tiap paket yang dikirimkan. Hal ini membantu penerima untuk melakukan

antisipasi jika terjadi paket yang hilang. RTP adalah protokol yang dibuat untuk mengkompensasi *jitter* dan *desequencing* yang terjadi pada jaringan IP. RTP dapat digunakan untuk beberapa informasi tipe data yang dikirim, *timestamps* yang digunakan untuk beberapa macam data *stream* yang *real-time* seperti data suara dan data video.

Datagram RTP, Protokol RTP berisi informasi tipe data yang dikirim, *timestamps* yang digunakan untuk pengaturan waktu dan suara percakapan terdengar seperti sebagaimana diucapkan, dan *sequence number* yang digunakan untuk pengurutan paket data dan mendeteksi adanya paket hilang. RTP didesain untuk digunakan pada *transport layer*, namun demikian RTP digunakan diatas UDP, bukan pada TCP karena TCP tidak dapat beradaptasi pada pengiriman data yang *real-time* dengan keterlambatan yang relatif kecil seperti pada pengiriman data komunikasi suara. Dengan UDP dapat mengirimkan paket IP secara *multicast*, RTP *stream* yang dibentuk oleh suatu terminal dapat dikirimkan ke beberapa terminal tujuan.

#### 2.1.5.3 Real-Time Streaming Protocol (RTSP)

*Real-Time Streaming Protocol* (RTSP) adalah protokol tingkat aplikasi untuk mengontrol multimedia data secara *real-time*. RTSP mendukung perangkat interaktif, seperti tombol-tombol standar pada sebuah video player, seperti: *play*, *pause* dan lainnya. Server *streaming* juga bisa memberi reaksi jika terjadi kemacetan jaringan, yaitu dengan mengubah *bandwidth* media untuk menggunakan kapasitas yang tersedia.

RTSP juga dikembangkan untuk menyamakan *syntax* dan operasi dari HTTP versi 1.1. Dengan menggunakan protokol RTSP antara *client* dan video *streaming server* bisa mengirimkan permintaan ketika terjadi interaksi, RTSP mempunyai kemampuan untuk mempertahankan keadaan dari sebuah sesi (mempertahankan hubungan antara video *streaming server* dan *client* secara konsisten), sedangkan protokol HTTP tidak bisa. RTSP mendukung penggunaan dari RTP sebagai dasar protokol pengiriman data. Protokol tersebut diharapkan bisa memberikan cara pemilihan jalur pengiriman yang optimum kepada *client*.

#### 2.1.5.4 Real-Time Control Protocol (RTCP)

RTCP digunakan sebagai penghubung dengan RTP. Protokol tersebut memberikan *feedback* kepada setiap penerima pada sesi RTP dimana bisa digunakan untuk mengontrol sebuah sesi. Pesan tersebut terdiri dari laporan, termasuk jumlah paket yang hilang dan statistik dari *jitter* (kedatangan yang terlalu cepat dan keterlambatan / *error* yang terjadi ). Informasi ini bisa kemacetan jaringan. Beberapa pesan RTCP berhubungan dengan pengontrolan konferensi video dengan beberapa penerima.

### 2.1.6 Live Video Streaming

Live streaming, aplikasi *live streaming* dapat dijumpai dalam teknologi *broadcast* radio dan televisi. Aplikasi ini memungkinkan pengguna untuk menerima siaran radio dan televisi secara langsung (*live*). Dalam *live streaming* tidak ada data video yang disimpan di dalam *server* sehingga *Client* tidak dapat melakukan proses *fast forward* dalam media yang diakses. Proses *capture* dan *encoding* secara langsung dilakukan sesuai dengan format videonya sebelum video itu ditransmisikan kepada *client*.

## 2.2 Wireless Local Area Network (WLAN)

*Wireless Local Area Network* adalah suatu jaringan area lokal nirkabel yang menggunakan gelombang radio sebagai media transmisinya, link terakhir yang digunakan adalah nirkabel, untuk memberi sebuah koneksi jaringan ke seluruh pengguna dalam area sekitar. Area dapat berjarak dari ruangan tunggal ke seluruh jaringan luas. Jaringan biasanya menggunakan kabel, dengan satu atau lebih titik akses jaringan menyambungkan pengguna nirkabel ke jaringan berkabel.

### 2.2.1 Spesifikasi WLAN 802.11

Wifi dirancang berdasarkan spesifikasi IEEE 802.11. Ada beberapa jenis spesifikasi dari 802.11 berdasarkan tingkat kecepatan yaitu 802.11a, 802.11b, 802.11g, dan 802.11n. Untuk spesifikasi lebih lanjut dapat di lihat pada Tabel 2.2 :

Tabel 2.2 Spesifikasi 802.11

| Tipe    | Kecepatan | Frekuensi |
|---------|-----------|-----------|
| 802.11a | 54 Mbps   | 5 Ghz     |
| 802.11b | 11 Mbps   | 2,4 Ghz   |
| 802.11g | 54 Mbps   | 2,4 Ghz   |
| 802.11n | 300 Mbps  | 2,4 Ghz   |

Sumber : <https://id.wikipedia.org/wiki/Wi-Fi>

#### 2.2.1.1 802.11a

Disahkan juga oleh IEEE pada tanggal 16 September 1999, 802. 11a memakai *OFDM*. Dengan kecepatan maksimum data 54 Mbps, dengan *throughput* sampai setinggi 27 Mbps. 802.11a beroperasi di ISM band antara 5.745 dan 5.805 GHz, dan bagian dari UNI band diantara 5.150 dan 5.320 GHz. Ini membuatnya tidak cocok dengan 802.11b atau 802.11g, dan frekuensi yang lebih tinggi berarti jangkauannya lebih pendek dari pada 802.11b/g dengan daya pancar yang sama. Memang bagian dari spektrumnya relatif tidak dipakai dibandingkan dengan 2.4 GHz, sayangnya dia hanya legal digunakan di sedikit negara di dunia. Tanyakan kepada pihak yang berwenang sebelum memakai peralatan

802.11a, terutama untuk penggunaan di luar ruangan. Peralatan 802.11a sebetulnya relatif murah, tapi tidak sepopuler 802.11 b/g.

#### 2.2.1.2 802.11b

Disahkan oleh *IEEE* pada tanggal 16 September 1999, 802.11b mungkin adalah protokol jaringan nirkabel yang paling populer yang dipakai saat ini. Jutaan alat-alat untuk mendukungnya telah dikeluarkan sejak 1993. 802.11b memakai modulasi yang dikenal sebagai *Direct Sequence Spread Spectrum* (DSSS) bagian dari ISM band dari 2.400 sampai 2.495 GHz. Mempunyai kecepatan maksimum 11 Mbps, dengan kecepatan sebenarnya yang bisa dipakai sampai 5 Mbps.

#### 2.2.1.3 802.11g

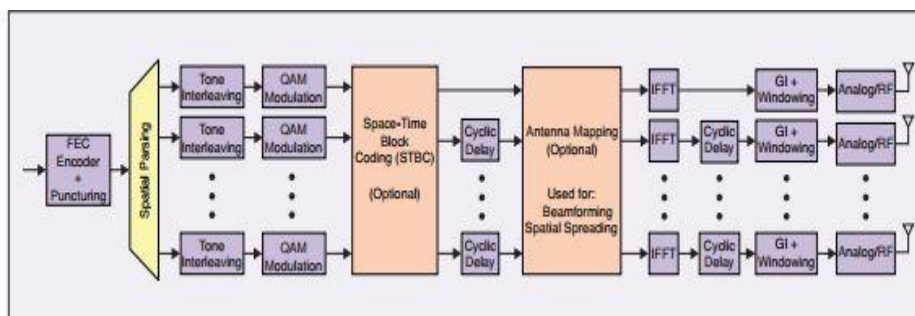
Digunakan mulai pertengahan 2003 dengan menggunakan frekuensi 2,4 GHz. Maksimum *bandwidth* yang bisa dicapai sebesar 54 Mbps. Modulasi yang digunakan adalah *OFDM*. Kanal yang tidak *overlapping* berjumlah tiga buah. Protokol ini kompatibel dengan tipe 802.11b. IEEE 802.11g adalah sebuah standar jaringan nirkabel yang bekerja pada frekuensi 2,45 GHz dan menggunakan metode modulasi *OFDM*. 802.11g yang dipublikasikan pada bulan Juni 2003 mampu mencapai kecepatan hingga 54 Mb's pada pita frekuensi 2,45 GHz, sama seperti halnya IEEE 802.11 biasa dan IEEE 802.11b. Standar ini menggunakan modulasi sinyal *OFDM*, sehingga lebih resistan terhadap interferensi dari gelombang lainnya.

#### 2.2.1.4 802.11n

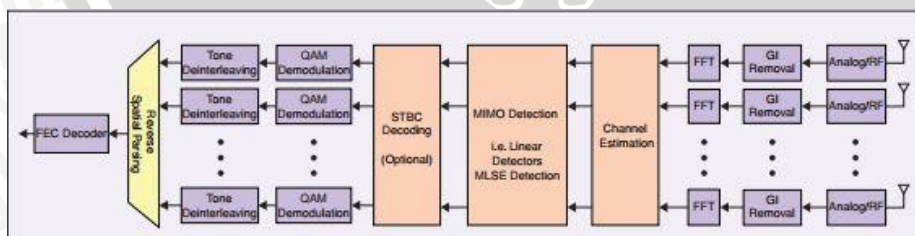
IEEE 802.11n-2009 adalah sebuah perubahan standar jaringan nirkabel 802.11-2.007 IEEE untuk meningkatkan throughput lebih dari standar sebelumnya, seperti 802.11b dan 802.11g, dengan peningkatan data rate maksimum dalam lapisan fisik OSI (PHY) dari 54 Mbps ke maksimum 300 Mbps dengan menggunakan empat ruang aliran di lebar channel 40 MHz. Sejak tahun 2007, Wi-Fi Alliance telah memberikan sertifikat interoperabilitas produk “draft-n” berdasarkan pada draft 2.0 dari spesifikasi IEEE 802.11n. Aliansi telah meningkatkan perangkat ini dengan tes kompatibilitas untuk beberapa perangkat tambahan yang diselesaikan setelah Draft 2.0. Lebih jauh lagi, telah ditegaskan bahwa semua produk bersertifikat draft-n tetap kompatibel dengan produk-produk standar terakhir.

IEEE 802.11n didasarkan pada standar 802.11 sebelumnya dengan menambahkan *multiple-input multiple-output* (MIMO) dan 40 MHz ke lapisan saluran fisik (PHY), dan frame agregasi ke MAC layer. MIMO adalah teknologi yang menggunakan beberapa antena untuk menyelesaikan informasi lebih lanjut secara koheren daripada menggunakan satu antena. Dua manfaat penting MIMO adalah menyediakan keragaman antena dan spasial

multiplexing untuk 802.11n. Kemampuan lain teknologi MIMO adalah menyediakan *Spatial Division Multiplexing* (SDM). SDM secara spasial me-multiplexes beberapa stream data independen, ditransfer secara serentak dalam satu saluran spektral bandwidth. MIMO SDM dapat meningkatkan throughput data seperti jumlah dari pemecahan stream data spasial yang ditingkatkan. Setiap aliran spasial membutuhkan antenna yang terpisah baik pada pemancar dan penerima. Di samping itu, teknologi MIMO memerlukan rantai frekuensi radio yang terpisah dan analog-ke-digital converter untuk masing-masing antenna MIMO yang merubah biaya pelaksanaan menjadi lebih tinggi dibandingkan dengan sistem non-MIMO. Saluran 40 MHz adalah fitur lain yang dimasukkan ke dalam 802.11n yang menggandakan lebar saluran dari 20 MHz di 802.11 PHY sebelumnya untuk mengirimkan data. Hal ini memungkinkan untuk penggandaan kecepatan data PHY melebihi satu saluran 20 MHz. Hal ini dapat diaktifkan di 5 GHz mode, atau dalam 2.4 GHz jika ada pengetahuan yang tidak akan mengganggu beberapa 802.11 lainnya atau sistem non-802.11 (seperti Bluetooth) menggunakan frekuensi yang sama. Arsitektur coupling MIMO dengan saluran bandwidth yang lebih luas menawarkan peningkatan fisik transfer rate melebihi 802.11a (5 GHz) dan 802.11g (2.4 GHz).



Gambar 2.2 Blok Diagram WLAN 802.11n Saat pengiriman  
Sumber : <https://Understanding the IEEE 802.11n Amendment/>



Gambar 2.3 Blok Diagram WLAN 802.11n Saat Menerima  
Sumber : <https://Understanding the IEEE 802.11n Amendment/>

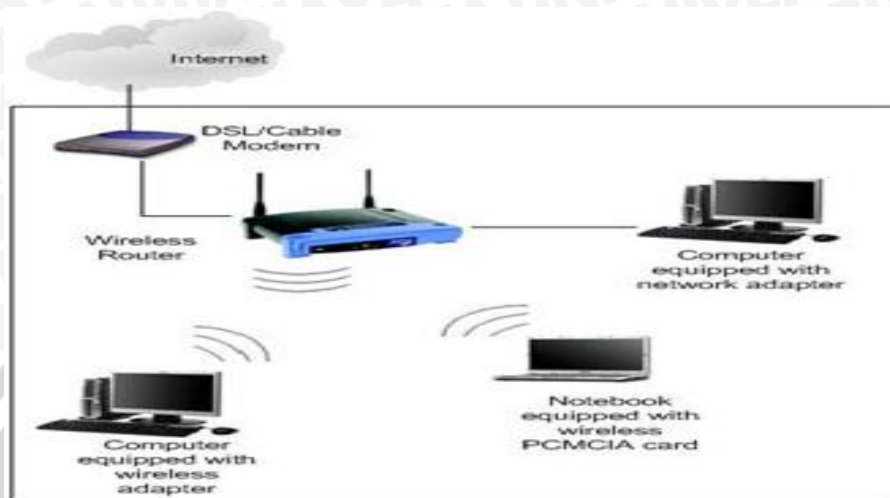
#### Keterangan :

1. Forward Error Correction : Suatu sistem dari kontrol kesalahan untuk transmisi data dimana pengirim secara sistematis yang dihasilkan menambahkan data yang berlebihan untuk pesan di dalamnya.
2. Spasial parsing : Teknik transmisi dalam komunikasi nirkabel MIMO untuk mengirimkan sinyal data independen dan terpisah lalu dikodekan.
3. Tone interleaving : Teknik yang memungkinkan penggunaan beberapa konverter analog - ke - digital identik untuk memproses sampel seri data reguler pada tingkat yang lebih cepat daripada tingkat sampel operasi masing-masing converter data individu.
4. QAM Modulation : Modulasi yang membawa data dengan mengubah (memodulasi) amplitudo dari dua gelombang pembawa. Kedua gelombang tersebut, biasanya sinusoid, berbeda fase dengan yang lainnya sebesar  $90^\circ$ .
5. Space Time Block Coding : Pembatalan sinyal yang tidak diinginkan akan terjadi ketika saluran datar dan simbol data yang sama yang dikirimkan melalui beberapa antena.
6. Cyclic Delay : Delay yang terjadi karena pergeseran fasa.
7. Antenna Mapping : Digunakan Apabila power amplifier tidak optimal bekerja.
8. Inverse Fast Fourier Transform : Menyatukan Sinyal Diskrit.
9. Guard Interval : Memastikan bahwa transmisi yang berbeda tidak mengganggu satu sama lain.
10. Windowing : Metode untuk mengendalikan kepadatan trafik dimana sebuah window ditentukan oleh sistem penerima untuk membatasi jumlah segmen data (bytes) yang dapat dikirimkan oleh perangkat source tanpa adanya ACK dari penerima.
11. Analog RF : Mengirim dan menerima gelombang elektromagnetik.

#### 2.2.2 Prinsip Kerja WLAN

Jaringan *Wireless Local Area Networks* (WLAN) yang didasari pada spesifikasi IEEE 802.11. Standar terbaru dari spesifikasi 802.11a atau b, seperti 802.11g, 802.11n saat ini sedang dalam penyusunan, spesifikasi terbaru tersebut menawarkan banyak peningkatan mulai dari luas cakupan yang lebih jauh hingga kecepatan transfernya.

Awalnya wifi ditujukan untuk penggunaan perangkat nirkabel dan Jaringan Area Lokal (LAN), namun saat ini lebih banyak digunakan untuk mengakses internet. Hal ini memungkinkan seseorang dengan komputer dengan kartu nirkabel (*wireless card*) atau *personal digital assistant* (PDA) untuk terhubung dengan internet dengan menggunakan titik akses (atau dikenal dengan hotspot) terdekat.



Gambar 2.4 Prinsip Kerja WLAN

Sumber : <http://jaringankomputer.org/wireless-lan/>

Wireless LAN bekerja dengan menggunakan gelombang radio. Sinyal radio menjalar dari pengirim ke penerima melalui *free space*, pantulan-pantulan, difraksi, *line of sight* dan *obstructed* tiap sinyal (pada jalur yang berbeda-beda) memiliki level kekuatan, *delay* dan fasa yang berbeda-beda. Mirip dengan jaringan *Ethernet* kabel, sebuah *wireless* LAN mengirim data dalam bentuk paket. Setiap adapter memiliki no ID yang permanen dan unik yang berfungsi sebagai sebuah alamat dan tiap paket selain berisi data juga menyertakan alamat penerima dan pengirim paket tersebut. Sama dengan sebuah adapter *Ethernet*, sebuah kartu, *wireless* LAN akan memeriksa kondisi jaringan sebelum mengirim paket ke dalamnya. Bila jaringan dalam keadaan kosong, maka paket langsung dikirimkan. Bila kartu mendeteksi adanya data lain yang sedang menggunakan frekuensi radio, maka ia menunggu sesaat kemudian memeriksanya kembali.

Teknologi utama yang banyak digunakan untuk membuat jaringan nirkabel adalah keluarga protokol 802.11, dikenal juga sebagai Wi-Fi. Sementara protokol-protokol baru seperti 802.16 (dikenal juga sebagai WiMax) seperti ini bias menyelesaikan beberapa kesulitan yang tampak pada 802.11.

## 2.3 Quality of Service Video Streaming

Terdapat beberapa faktor yang mempengaruhi kualitas layanan *Video Streaming*, yaitu : *delay*, *throughput*, *packet loss ratio*.

### 2.3.1 Delay End-to-End

*Delay End-to-End* adalah waktu yang dibutuhkan untuk mengirim sebuah paket dari sumber menuju ke tujuan (ujung ke ujung). Berdasarkan ITU G.114 membagi karakteristik waktu tunda berdasarkan tingkat kenyamanan user, dapat ditunjukkan pada Tabel 2.3 :

Tabel 2.3 Pengelompokan *Delay* Berdasarkan ITU-T G.114

| Waktu tunda (ms) | Kualitas                    |
|------------------|-----------------------------|
| 0-150            | Baik                        |
| 150-300          | Cukup, masih dapat diterima |
| >300             | Buruk                       |

Sumber: ITU-T G.114, 2000

Dalam jaringan berbasis *packet switching*, *delay* yang terjadi merupakan penjumlahan *delay-delay* yang ada dalam perjalanan paket dari sumber ke tujuannya pada setiap hop. Waktu tunda pengiriman paket data pada jaringan dirumuskan seperti dalam Persamaan (2-1).

$$t_{jaringan} = t_{proses} + t_{trans} + t_p + t_w \quad (2-1)$$

Keterangan:

- $t_{proses}$  = Delay proses(s)
- $t_{trans}$  = Delay transmisi (s)
- $t_p$  = Delay propagasi (s)
- $t_w$  = Delay antrian (s)

Sementara *delay end-to-end* untuk media layanan live streaming pada jaringan WLAN terdiri atas *delay codec* dan *delay jaringan WLAN* yang dirumuskan dalam Persamaan (2-2).

$$t_{total} = t_{codec} + t_{jaringan} \quad (2-2)$$

Keterangan :

- $t_{total}$  = Delay total
- $t_{codec}$  = Delay codec
- $t_{jaringan}$  = Delay end to end WLAN

Dan rumus delay untuk pengujian menggunakan Persamaan (2-3).

$$Delay = \frac{duration}{total\ packet} \quad (2-3)$$

Keterangan :

Duration = total waktu pengiriman paket

Total paket = total paket yang dikirim

### 2.3.1.1 Delay end-to-end pada Codec

*Delay end-to-end* aplikasi *live video streaming* merupakan jumlah *delay* CODEC aplikasi *live video streaming* dengan *delay* jaringan dimana aplikasi tersebut berjalan. *Delay* CODEC aplikasi *live video streaming* terdiri dari *delay* CODEC audio dan video untuk menghasilkan satu paket *video streaming*. Besarnya *delay* CODEC aplikasi *video streaming* dapat dihitung dengan menggunakan Persamaan (2-4).

$$t_{CODEC} = t_a + t_v \quad (2-4)$$

Keterangan :

$t_{CODEC}$  = *delay* CODEC aplikasi *video streaming* (detik)

$t_a$  = *delay* CODEC audio (detik)

$t_v$  = *delay* CODEC video (detik)

### 2.3.1.2 Delay end-to-end jaringan

*Delay* pada jaringan merupakan penjumlahan *delay-delay* yang ada dalam perjalanan paket dari sumber ke tujuannya pada setiap hop. *Delay* tersebut antara lain: *delay* proses, *delay* transmisi, *delay* propagasi, *delay* antrian.

#### 2.3.1.2.1 Delay Proses

*Delay* proses terdiri dari proses enkapsulasi dan dekapsulasi. Enkapsulasi adalah proses menambahkan *header* pada paket data, sehingga paket data tersebut dapat tepat sampai ke tujuan. *Delay* enkapsulasi adalah waktu yang dibutuhkan untuk menambahkan keseluruhan *header* pada sebuah paket. Sedangkan *delay* dekapsulasi adalah waktu yang dibutuhkan untuk melepaskan keseluruhan *header* dari sebuah paket.

Apabila *node* sumber ingin mengirim paket data ke *node* tujuan, maka proses yang terjadi adalah paket tersebut dikirimkan ke *transport layer*. Di *transport layer* dengan menggunakan protokol TCP, paket data dienkapsulasi menjadi segmen TCP. *Delay* enkapsulasi ditunjukkan pada Persamaan (2-5) dan *delay* dekapsulasi ditunjukkan oleh Persamaan (2-6).

$$t_{enc/dec} = \frac{L_{Header\ Ipv4} + L_{Header\ NALU} + L_{Header\ UDP} + L_{Header\ RTP} + L_{Header\ Ethernet} + L_{Header\ FCS}}{C_{proses}} \times 8 \quad (2-5)$$

$$C_{proses} = \frac{N_k}{T_k} \quad (2-6)$$

Keterangan:

$t_{enc/dec}$  = *delay* enkapsulasi dan *delay* dekapsulasi

$C_{pros}$  = kecepatan pemrosesan pada terminal pengirim (bps);

$N_k$  = jumlah total data yang dikirimkan (bit);

$T_k$  = waktu pengiriman total data (s)

Tabel 2.4 Header Format

| Protokol               | Kuantitas | Sumber                           |
|------------------------|-----------|----------------------------------|
| RTP Header             | 12 byte   | Peterson, L, <i>et al</i> , 2007 |
| IPv4 Header            | 20 byte   | RFC 791,                         |
| UDP Header             | 8 byte    | Stalling, 2007                   |
| Cyclic Redudancy Check | 4 byte    | Forouzan, 2007                   |
| Ethernet Header        | 14 byte   | Forouzan, 2007                   |

### 2.3.1.2 Delay Transmisi

*Delay transmisi* adalah waktu yang diperlukan paket data untuk melalui media transmisi dari satu node ke node yang lain. Media transmisi bisa melalui serat optik, jaringan lokal tembaga maupun air interface. *Delay* transmisi ditentukan dengan Persamaan (2-7) (Forouzan, 2007).

$$t_t = \frac{(L+L')}{c} \quad (2-7)$$

Keterangan:

$t_t$  = *Delay* transmisi (s)

$L$  = panjang paket data (byte/paket)

$L'$  = panjang *header* (byte/paket)

$C$  = kecepatan transmisi (bps)

### 2.3.1.3 Delay Propagasi

*Delay propagasi* adalah waktu yang dibutuhkan oleh paket data dalam bentuk gelombang elektromagnetik untuk merambat dengan media transmisi dari *Node* sumber menuju tujuan. *Delay* propagasi ditentukan dengan Persamaan (2-8) (Forouzan, 2007:90).

$$t_p = \frac{d_{max}}{v} \quad (2-8)$$

Keterangan:

$t_p$  = *Delay* propagasi (s)

$d_{max}$  = Jarak *Access Point* terhadap *client* (m)

$v$  = Cepat rambat gelombang elektromagnetik ( $3 \times 10^8$  m/s)

### 2.3.1.4 Delay Antrian

*Delay antrian* adalah rentang waktu paket data berada dalam antrian untuk ditransmisikan. Model antrian yang digunakan adalah model *First In First Out* (FIFO), yakni data pertama yang masuk akan diproses terlebih dahulu dan akan dikeluarkan terlebih dahulu. Besarnya delay antrian tersebut ditentukan oleh Persamaan (2-9) (Mischa Schwartz, 1987: 42).

$$t_w = \frac{1}{\mu} + \frac{\lambda_w}{\mu^2(1-\rho)} \quad (2-9)$$

Keterangan :

- $t_w$  = delay antrian (ms)
- $\lambda_w$  = kecepatan kedatangan (paket/s)
- $\mu$  = kecepatan pelayanan (paket/s)
- $\rho$  = faktor utilitas sistem *wireless*

### 2.3.2 Packet Loss

*Packet loss* adalah jumlah paket data yang hilang saat proses transmisi terjadi. *Packet loss* dapat disebabkan oleh sejumlah faktor, mencakup penurunan signal dalam media jaringan, melebihi batas saturasi jaringan, paket yang *corrupt* yang menolak untuk transit, dan kesalahan keras jaringan. Tabel 2.4 memperlihatkan standar tingkat paket hilang pada jaringan:

Tabel 2.5 Standar tingkat paket hilang berdasarkan ITU-T-G1010

| Tingkat Paket Hilang | Kualitas |
|----------------------|----------|
| 0 – 5 %              | Baik     |
| 5 – 10 %             | Cukup    |
| > 10 %               | Buruk    |

Sumber: ITU-T G.1010, 2002

Probabilitas *packet loss* merupakan banyaknya kemungkinan paket yang diterima mengalami kesalahan pada sisi penerima. Probabilitas *packet loss* video streaming pada suatu jaringan ditentukan berdasarkan pada probabilitas *packet loss* pada jaringan tersebut serta probabilitas *packet loss* video streaming yang berbasis protokol UDP/RTP/IP ditunjukkan pada Persamaan (2-10) (Stallings, 1997: 513).

$$\rho_{total} = 1 - [(1 - \rho_{network})(1 - \rho_{UDP/RTP/IP})] \quad (2-10)$$

Keterangan :

- $\rho_{total}$  = Probabilitas *packet loss* total  
 $\rho_{network}$  = Probabilitas *packet loss* pada WLAN 802.11n  
 $\rho_{UDP/RTP/IP}$  = Probabilitas *packet loss* video streaming

dimana:

$$P_{network} = \frac{e^{-\frac{Eb}{No}}}{2 \sqrt{\pi \left(\frac{Eb}{No}\right)}} \quad (2-11)$$

Keterangan :

- $\rho_{network}$  = Probabilitas *packet loss* pada WLAN 802.11n  
 $Eb$  = Energy per bit  
 $No$  = Noise level pada WLAN 802.11n

Probabilitas *packet loss* dalam persen dirumuskan dalam persen dirumuskan pada Persamaan (2-13).

$$\rho_{total}(\%) = \rho_{total} \times 100\% \quad (2-12)$$

Sementara Rumus yang digunakan untuk menghitung probabilitas *packet loss* dalam pengujian ini menggunakan Persamaan (2-14).

$$Packet Loss = \frac{Packet\ Terkirim - Packet\ Diterima}{Packet\ Terkirim} \times 100\% \quad (2-13)$$

Keterangan :

- Paket terkirim = Total UDP paket yang terkirim  
 Paket diterima = Paket yang berhasil diterima

### 2.3.3 Throughput

*Throughput* adalah jumlah rata-rata paket yang sukses diterima atau dikirimkan oleh saluran penerima atau pemancar per detik. *Throughput* merupakan salah satu parameter yang menunjukkan kinerja dari suatu sistem komunikasi data. *Throughput* dapat dihitung dengan menggunakan Persamaan (2-10) (Mischa Schwartz, 2005:411)

$$\gamma = \frac{(1-\rho)}{t_1[1+(\alpha-1)\rho]} \quad (2-14)$$

Keterangan :

- $\gamma$  = *throughput* pada sistem (bps)  
 $\rho$  = probabilitas *packet loss* yang diterima  
 $\alpha$  = konstanta propagasi  
 $t_1$  = waktu transmisi sebuah paket (s)

dimana:

$$\alpha = (1 + \frac{2t_{prop} + 2t_{trans}}{t_{trans}}) \quad (2-15)$$

Sementara dalam pengujian ini rumus yang digunakan untuk menghitung *throughput* dapat dilihat pada Persamaan (2-16).

$$Throughput = \frac{\text{Jumlah bit yang dikirim}}{\text{Total Waktu Pengiriman paket}} \quad (2-16)$$

Beberapa faktor yang menentukan nilai *throughput* adalah :

1. Piranti jaringan.
2. Tipe data yang dikirim.
3. Banyaknya pengguna jaringan
4. Spesifikasi komputer *server* dan *client*.

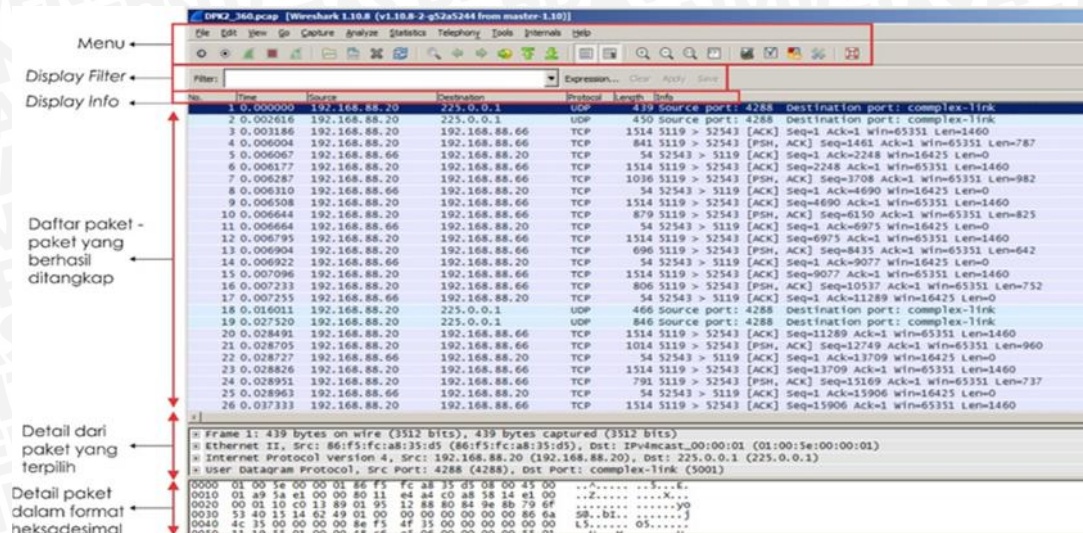
## 2.4 Perangkat Lunak (Software) Pendukung

Pada skripsi ini adapun perangkat lunak yang mendukung dalam pengujian ini adalah *Wireshark* dan Video LAN Client. Kedua perangkat lunak ini bersifat *open source* dan tersedia dalam operating sistem windows maupun linux.

### 2.4.1 Wireshark

*Wireshark* adalah aplikasi gratis yang dapat menangkap paket-paket jaringan (*sniffing*) pada sisi *client* yang berguna untuk menampilkan informasi secara detail pada paket tersebut. Fungsi utamanya adalah membantu administrator jaringan dalam menganalisis kinerja jaringan dan protokol yang terkandung didalamnya. *Wireshark* bekerja pada *Application layer*. *Wireshark* mempunyai banyak fitur dan kelebihan diantaranya adalah sebagai berikut:

1. Aplikasi *Wireshark* bersifat *open source* untuk menganalisis paket jaringan.
2. Mampu menangkap paket data secara langsung dari sebuah *network interface*.
3. Mampu menampilkan informasi secara detail mengenai hasil *capture* pada sebuah jaringan.
4. Mampu menampilkan hasil statistika dari hasil *capture* pada sebuah jaringan.
5. Tersedia untuk Linux dan Windows.



Gambar 2.6 Contoh Tampilan Wireshark yang Sedang Meng-capture Paket-Paket Pada Suatu Sistem Jaringan  
Sumber : Perancangan

Keterangan:

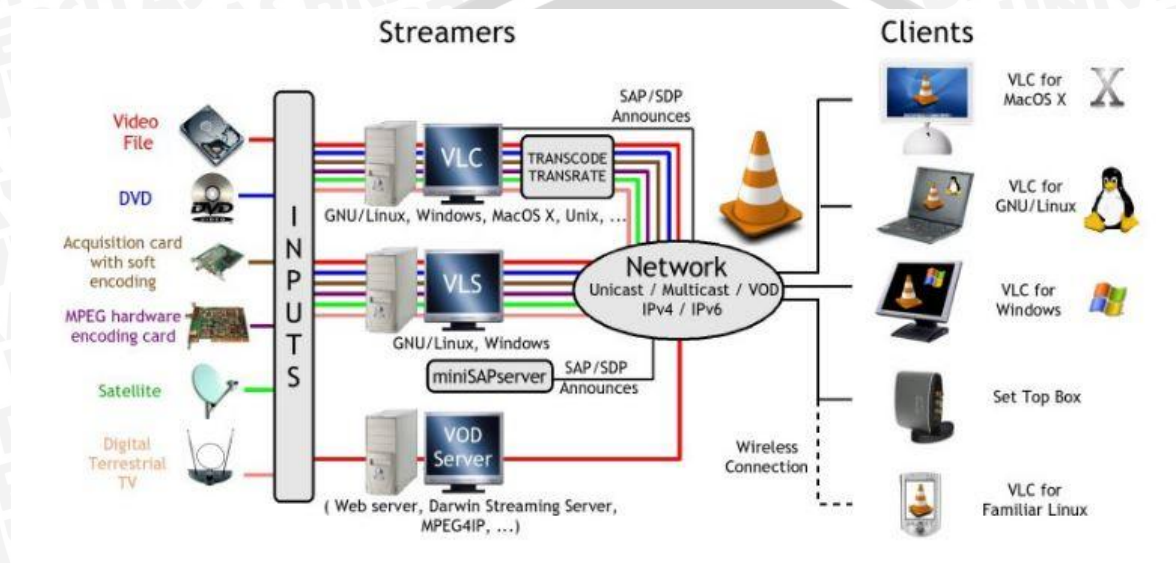
1. **Menu** : Menu-menu yang tersedia di *Wireshark*
2. **Display Filter** : Sebuah kolom yang dapat diisi dengan sintaks-sintaks untuk membatasi paket-paket apa saja yang akan ditampilkan pada *list* paket.
3. **Daftar Paket** : Menampilkan paket-paket yang berhasil ditangkap oleh Aplikasi *Wireshark*, berurutan dari paket pertama yang ditangkap, dan seterusnya.
4. **Detail Paket** : Menampilkan detail paket yang terpilih pada daftar paket.
5. **Detail Heksa** : Menampilkan detail paket yang terpilih yang ditampilkan dalam bentuk heksa.

Pada daftar bagian daftar paket, terdapat kolom-kolom seperti berikut ini:

1. **Time** : Menampilkan waktu saat paket-paket tersebut ditangkap.
2. **Source** : Menampilkan alamat IP sumber dari paket data tersebut.
3. **Destination** : Menampilkan alamat IP tujuan dari paket data tersebut.
4. **Protocol** : Menampilkan protokol yang digunakan pada sebuah paket data.
5. **Info** : Menampilkan informasi secara detail tentang paket data tersebut.

### 2.4.2 Video LAN Client (VLC)

Video LAN Client (VLC) merupakan suatu media yang diperuntukkan bagi streaming, yang dapat dimanfaatkan sebagai aplikasi pengirim dan penerima. VLC ini dapat mengompresi dan dekompresi data audio maupun video dari beberapa media input dan mampu menunjukkan hasil *streaming* dengan kualitas yang sama dengan aslinya. Berikut merupakan blok diagram video streaming menggunakan VLC media player yang ditunjukan gambar 2.7



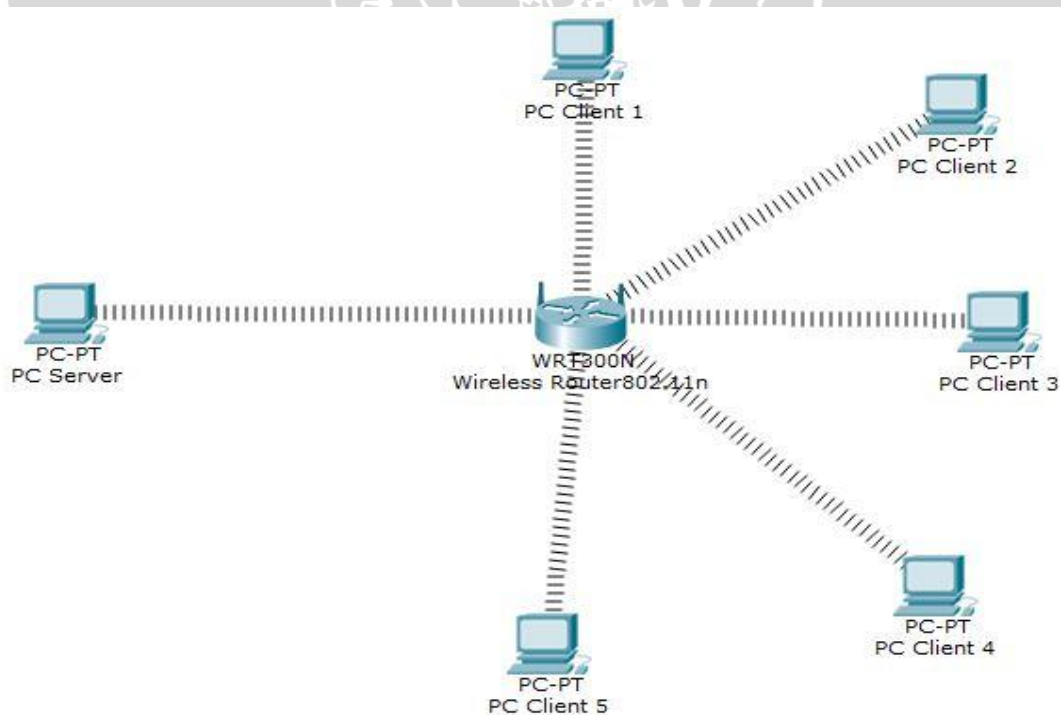
Gambar 2.7 Blok Diagram streaming video menggunakan VLC  
Sumber : <http://www.videolan.org/vlc/>

## BAB III

### METODE PENELITIAN

#### 3.1 Konfigurasi Jaringan

Penyusunan skripsi ini didasarkan pada masalah yang bersifat aplikatif, yaitu perencanaan dan perealisasiannya. Pada bab ini akan dibahas mengenai perancangan infrastruktur yang mendukung layanan *live streaming*, dan konfigurasi jaringan. Infrastruktur yang mendukung dalam skripsi ini antara lain adalah 6 buah laptop. 1 buah laptop bertindak sebagai *server*. Berdasarkan ITU-T G.114 jaringan dikatakan baik apabila nilai *delay end to end* kurang dari 150 ms, sementara itu berdasarkan hasil perhitungan jaringan WLAN 802.11n mampu melayani layanan *live streaming* dengan jumlah *client* 13 dengan nilai *delay end to end* kurang dari 150 ms tetapi dikarenakan keterbatasan perangkat maka jumlah *client* yang digunakan dalam skripsi ini berjumlah 5 *client*. Dan 1 buah *wireless router* dengan spesifikasi WLAN 802.11n. Dipilihnya WLAN 802.11n sebagai jaringan dalam penelitian ini dikarenakan saat ini jaringan tersebut banyak digunakan. Bentuk umum konfigurasi jaringan ditunjukkan pada Gambar 3.1.



Gambar 3.1 Konfigurasi Jaringan  
Sumber: Perancangan

Pada uji coba jaringan dengan menggunakan VLC sebagai *server streaming* dan *client streaming*. Selanjutnya pada sisi *client* digunakan *software wireshark* sebagai *network analyzer*. Berdasarkan hasil perhitungan, semakin dekat *client* dengan *wireless router* maka performansi layanan *live streaming* semakin baik yaitu pada jarak 1 meter. Namun dalam skripsi ini jarak antara *wireless router* dengan *client* 3 meter, tetapi dengan jarak 3 meter layanan *live streaming* pada jaringan WLAN 802.11n dikategorikan baik karena memiliki nilai *delay end to end* kurang dari 150 ms. Proses *streaming* dijalankan dengan beberapa tahap, tahapan-tahapan yang berlangsung ialah 2 *client* melakukan proses *streaming* dalam waktu yang bersamaan. Sementara selang waktu 1 menit 2 *client* melakukan proses *streaming*, dan 1 *client* melakukan proses *streaming* setelah selang waktu 2 menit. Dilakukannya proses *streaming* secara bertahap untuk menciptakan *traffic* pengiriman pada alamat IP *multicast*. Pengambilan data dilakukan di dalam ruangan (*indoor*) dengan kondisi *indoor* sinyal dapat terpantulkan dan terhalang dengan barang-barang yang ada di dalam ruangan. Pengambilan data dilakukan di ruang laboratorium saluran transmisi dan gelombang mikro.

### 3.2 Pengambilan Data

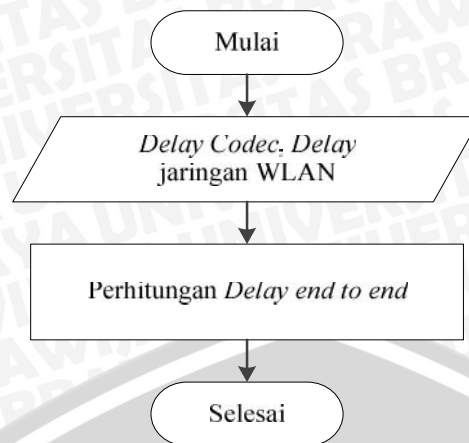
Pengambilan data dilakukan untuk mendapatkan data sekunder dan data primer. Data sekunder didapatkan dari kegiatan studi literatur buku, jurnal ilmiah, dan forum-forum resmi yang membahas WLAN 802.11n dan *live video streaming*. Data primer didapatkan dari hasil pengukuran terhadap kinerja sistem.

#### 3.2.1 Pengambilan Data Sekunder

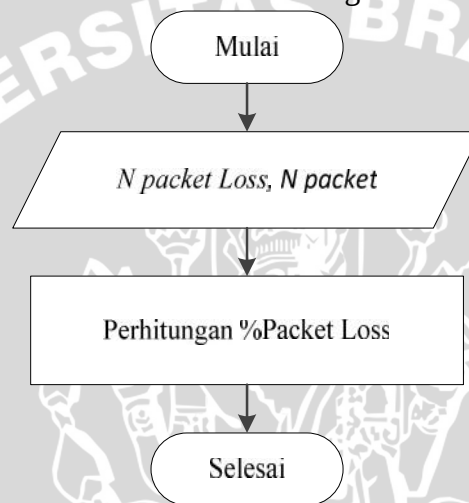
Pengambilan data sekunder dilakukan dengan cara melakukan studi literatur dengan referensi jurnal ilmiah, buku, dan forum-forum resmi. Studi literatur yang dilakukan bertujuan untuk mengkaji hal-hal yang berhubungan dengan teori-teori yang mendukung dalam perencanaan dan perealisasiian alat. Langkah-langkah yang dilakukan untuk mendapatkan data sekunder adalah sebagai berikut:

1. Mempelajari konsep dasar dan konfigurasi jaringan *Wireless Local Area Network* (WLAN).
2. Mempelajari konsep dasar dan konfigurasi layanan *Live Video Streaming*.
3. Mempelajari konsep dan parameter-parameter kinerja jaringan WLAN 802.11n.

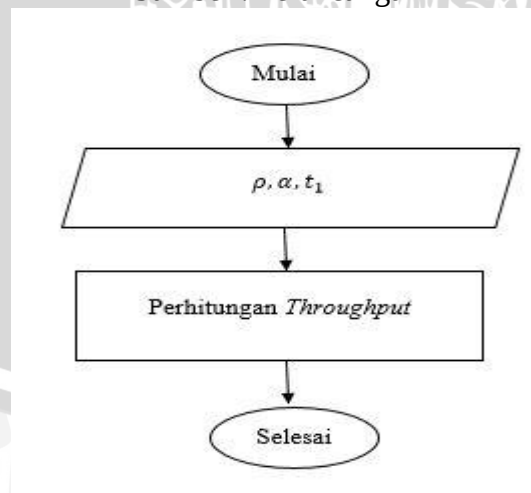
Berikut diagram alir untuk mendapatkan data sekunder untuk tiap-tiap parameter kinerja jaringan meliputi: *Delay End-to-End* (Gambar 3.2), *Packet Loss* (Gambar 3.3), *Throughput* (Gambar 3.4).



Gambar 3.2 Diagram Alir Perhitungan *Delay end-to-end*  
Sumber : Perancangan



Gambar 3.3 Diagram Alir Perhitungan *Packet Loss*  
Sumber : Perancangan



Gambar 3.4 Diagram Alir Perhitungan *Throughput*  
Sumber : Perancangan

### 3.2.2 Pengambilan Data Primer

Data primer adalah data yang didapatkan dari hasil pengamatan kinerja sistem yang dibuat. Kinerja sistem yang diukur melalui beberapa parameter QoS didapatkan dari hasil analisis perangkat lunak Wireshark yang dipasang pada sisi pengguna. Untuk melakukan pengambilan data primer hasil kinerja sistem dilakukan tahap-tahap sesuai Gambar 3.5.

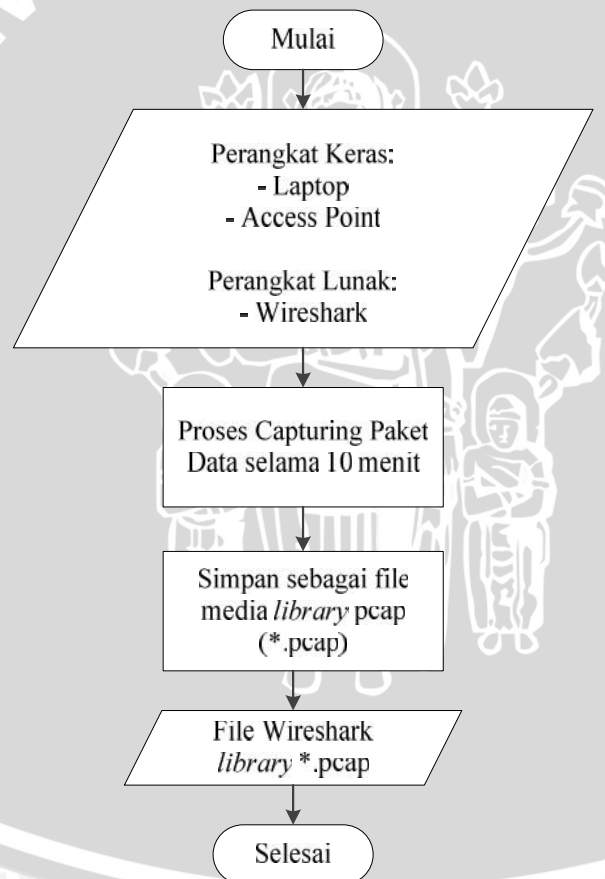


Gambar 3.5 Diagram Alir Pengambilan Data Primer  
Sumber : Perancangan

Pengambilan data dilakukan dengan pengiriman file video streaming dengan menggunakan aplikasi VLC yang sudah dikonfigurasi, Parameter yang dicatat adalah waktu pengiriman, jumlah dan besar paket yang diterima, dan transfer rate dari proses pengiriman (*throughput*). Parameter ini diamati dengan bantuan aplikasi WIRESHARK. Pada penelitian ini dilakukan pengamatan dan analisa terhadap QoS dari Video *streaming* dengan menggunakan pengaturan *Frame Rate* dengan mengubah nilainya yaitu mulai dari 10 fps, 25 fps, 40 fps, 60 fps. Serta mengubah *bitrate*, mulai dari 256 Kbps, 512 Kbps, 1024 Kbps. Pengamatan dilakukan dalam beberapa analisa yaitu:

1. Pengamatan dan analisa *delay* dari video *streaming* dengan pengaturan *frame rate* mulai dari 10 fps, 25 fps, 40 fps, 60 fps. Selanjutnya dilakukan pengamatan dengan mengubah *bitrate* 256 Kbps, 512 Kbps, 1024 Kbps.
2. Pengamatan dan analisa *throughput* dari video *streaming* dengan pengaturan *frame rate* mulai dari 10 fps, 25 fps, 40 fps, 60 fps. Selanjutnya dilakukan pengamatan dengan mengubah *bitrate* 256 Kbps, 512 Kbps, 1024 Kbps.
3. Pengamatan dan analisa *packet loss ratio* dari video *streaming* dengan pengaturan *frame rate* mulai dari 10 fps, 25 fps, 40 fps, 60 fps. Selanjutnya dilakukan pengamatan dengan mengubah *bitrate* 256 Kbps, 512 Kbps, 1024 Kbps.

Pengambilan data hasil kinerja sistem menggunakan perangkat lunak *Wireshark* yang dipasang pada sisi pengguna. Pengambilan data hasil kinerja sistem menggunakan *Wireshark* dilakukan sesuai Gambar 3.6.



Gambar 3.6 Proses pengambilan data menggunakan wireshark  
Sumber : Perancangan

Keluaran yang didapat dari hasil penggunaan *Wireshark* untuk pengambilan data primer adalah nomor *frame*, urutan waktu, *source* IP (alamat asal), *destination* IP (alamat tujuan), panjang paket (*length*), dan keterangan (info). Data primer dibentuk menjadi sebuah file *wireshark* dengan ekstensi media library pcap (\*.pcap). Data primer yang didapat selanjutnya akan dianalisis dan diolah pada langkah berikutnya.

### 3.3 Analisis Data

Analisis data bertujuan untuk menentukan kebutuhan data yang diperlukan. Setiap data yang didapatkan dari kegiatan pengambilan data akan disaring dan ditentukan kebutuhannya sesuai dengan pembahasan yang akan dilaksanakan setelahnya. Data yang akan dianalisis adalah data primer dan data sekunder. Luaran yang didapat dari analisis data untuk data primer adalah *file* data primer berekstensi \*.pcap yang telah didekodekan *Wireshark* sebagai protokol RTP. Dekode memungkinkan data untuk sementara mengalihkan pembedahan protokol tertentu. Setelah didekodekan maka *file* data primer tersebut dapat diolah dan dicari *RTP stream* yang terjadi. Analisis data sekunder menghasilkan pernyataan kesimpulan atas premis-premis pada data sekunder sebelumnya dan gambar analisa.

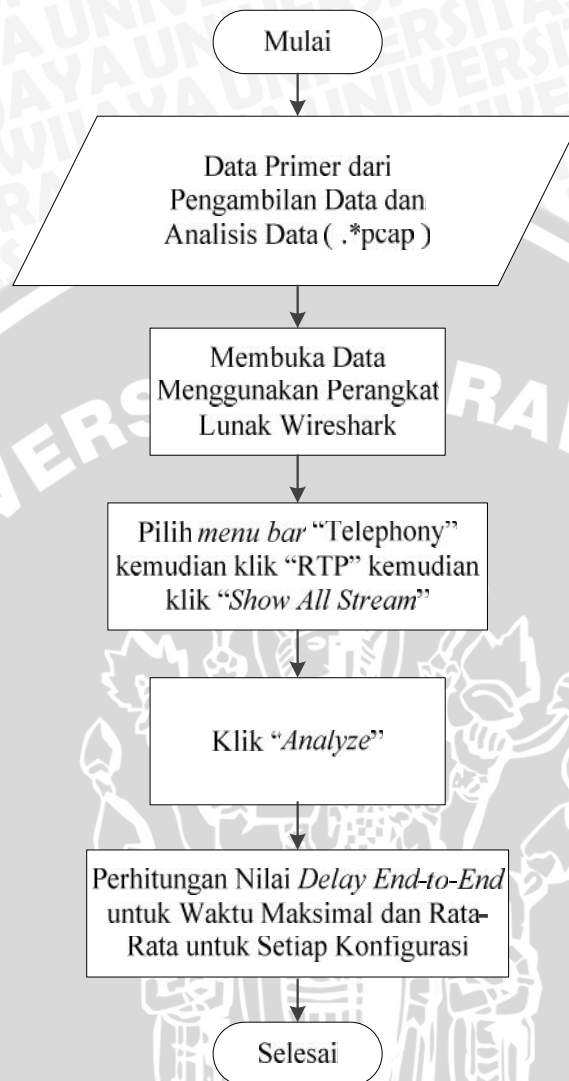
### 3.4 Pengolahan Data Primer

Data yang diolah pada kegiatan ini adalah data primer dan sekunder. Data primer yang didapatkan dari tahapan pengambilan data dan analisis data, kemudian diolah menjadi data primer yang berisi beberapa parameter QoS (*throughput*, *delay end-to-end*, *packet loss*) yang akan dibahas. Pada *Wireshark* terdapat *menu bar* “Telephony” dan “Statistics”. *Menu* “Telephony” berisi tentang protokol-protokol yang pada penelitian ini digunakan protokol RTP sehingga akan diolah aliran RTP-nya dan didapatkan parameter kinerja jaringannya. *Menu* “Statistics” berisi beberapa pilihan perintah dan salah satunya adalah “Summary” untuk menampilkan parameter *throughput*. Langkah dalam mengolah data primer yang diamati adalah sebagai berikut,

#### 3.4.1. Delay End-to-End

Tahap pengolahan data pada kinerja jaringan WLAN pada bagian *Delay End-to-End* adalah Pada *menu bar* “Telephony” terdapat pilihan RTP untuk menampilkan ringkasan percakapan atau komunikasi yang telah ditangkap dan menggunakan protokol RTP. Kemudian pilihan “Show RTP stream” membantu untuk menemukan aliran data dari sumber menuju tujuan maupun arah *reverse*. Pilihan “Analyze” membuat aliran RTP yang dipilih tadi diolah datanya dan menghasilkan informasi dan dimunculkan dengan kotak dialog RTP

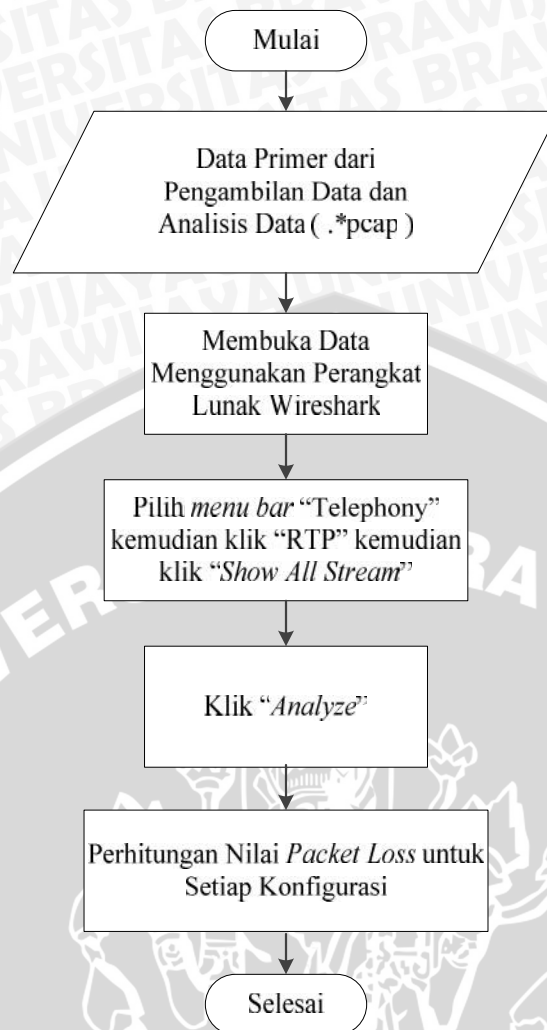
analysis. Pada kotak dialog RTP analysis terdapat berbagai macam informasi, yakni jumlah paket yang diterima atau kirim, *delta* (ms), *jitter* (ms) dan *Lost* (%) dan lain-lain. *Delay end-to-end* ditunjukkan oleh informasi *delta* (ms).



Gambar 3.7 Diagram alir pengolahan data *Delay End-to-End*  
Sumber : Perancangan

### 3.4.2 Packet Loss

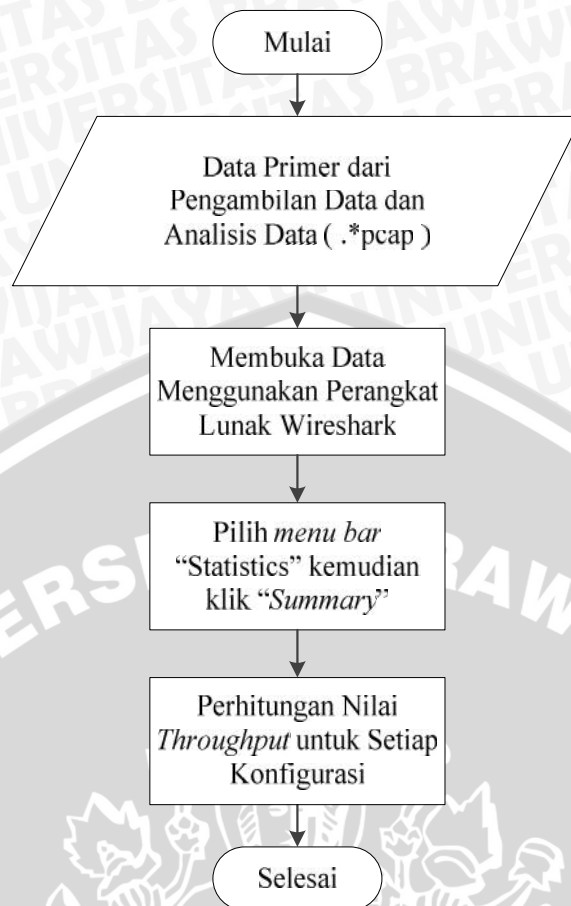
Tahap pengolahan data pada kinerja jaringan WLAN pada bagian *Packet Loss* adalah Pada kotak dialog RTP analysis terdapat berbagai macam informasi, yakni jumlah paket yang diterima atau kirim, *delta* (ms), *jitter* (ms) dan *Lost* (%) dan lain-lain. *Packet Lost* ditunjukkan oleh informasi *Lost* (ms).



Gambar 3.8 Diagram alir pengolahan data *Packet Loss*  
Sumber : Perancangan

### 3.4.3 Throughput

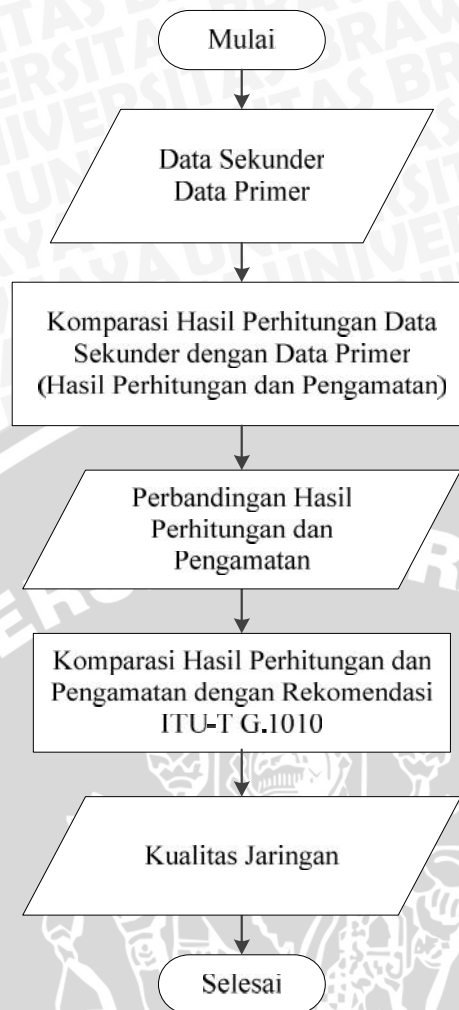
Tahap pengolahan data pada kinerja jaringan WLAN pada bagian *Throughput* adalah *File media library* pcap yang telah didapatkan dari tahapan pengambilan data primer dan analisis data kemudian dibuka kembali menggunakan perangkat lunak Wireshark. Pada Wireshark terdapat beberapa menu bar, yakni *File*, *Edit*, *Capture*, *Statistics*, *Interfaces*, *Analyze*, *Telephony*, *Help* dan *Window*. Pada menu bar "Statistics" terdapat pilihan *Summary* untuk menampilkan ringkasan percakapan atau komunikasi yang telah ditangkap. Pada kotak dialog *Summary* terdapat berbagai macam informasi, yakni *Time first packet into last packet*, *Avg Mbit/sec* dan lain-lain. Waktu paket pertama menuju paket terakhir adalah waktu keseluruhan *capturing* paket data. *Throughput* ditunjukkan oleh informasi *Avg Mbit/sec*



Gambar 3.9 Diagram alir pengolahan data *Throughput*  
Sumber : Perancangan

### 3.5 Pembahasan dan Hasil

Pembahasan membahas data sekunder dan data primer, yakni perhitungan secara teoretis parameter-parameter jaringan dengan hasil pengamatan dan perbandingan dengan rekomendasi atau syarat yang dikeluarkan ITU-T. G.1010 Sehingga dapat diketahui kualitas layanan *live streaming* yang dihasilkan pada sistem dan kualitas jaringan WLAN untuk layanan *live streaming*. Langkah yang dilakukan dalam pembahasan terdiri dari langkah-langkah sebagai berikut,



Gambar 3.10 Pembahasan dan Hasil  
Sumber : Perancangan

### 3.6 Penarikan Kesimpulan dan Saran

Kesimpulan diambil berdasarkan pembahasan dan hasil dari hasil perhitungan kinerja jaringan sesuai dengan metode dan tinjauan pustaka yang digunakan serta mengacu pada rumusan masalah yang ada. Pembahasan yang telah dilakukan, antara lain:

1. Hasil pengukuran kualitas jaringan *Wireless Local Area Network* pada layanan *live streaming* melalui Wireshark
2. Hasil pengukuran kualitas jaringan melalui perhitungan teoritis.
3. Hasil perbandingan kualitas jaringan yang dilewatkan layanan *live streaming* pada penelitian yang berbasis WLAN dari data primer dengan rekomendasi ITU-T G.1010

Setelah kesimpulan diambil maka saran akan digunakan untuk memperbaiki penelitian dan skripsi ini untuk perbaikan dan pengembangan ke tahap selanjutnya.

## BAB IV

### HASIL DAN PEMBAHASAN

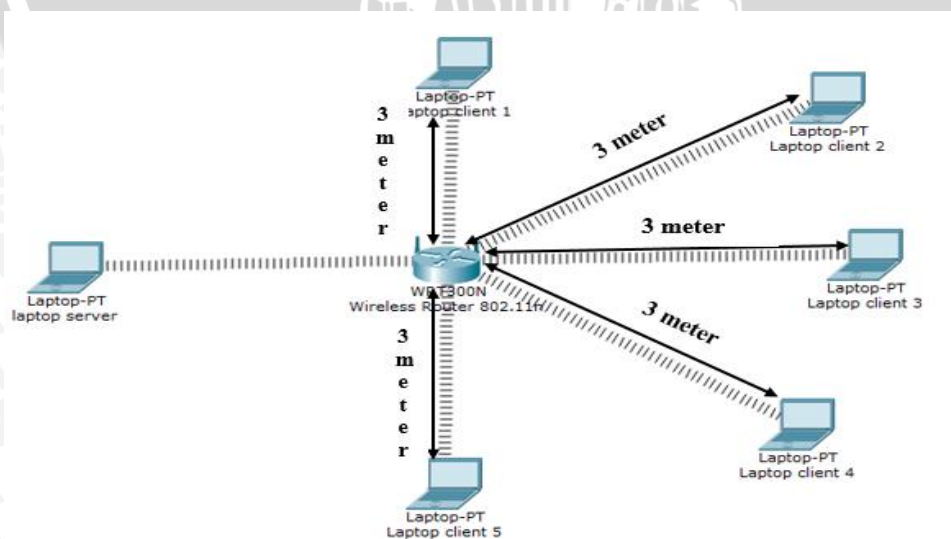
#### 4.1 Umum

Pembahasan yang dilakukan pada penelitian ini mengenai perancangan, instalasi jaringan hingga analisis kualitas jaringan WLAN untuk layanan *live streaming* yang dirancang pada jaringan *Wireless Local Area Network* (WLAN). Adapun beberapa tahap pembahasan dan analisis yang dilakukan:

- 1) Perancangan, instalasi, hingga pengujian sistem
- 2) Melakukan perhitungan dan pengambilan data kualitas jaringan WLAN untuk layanan *live streaming* yang telah dirancang, meliputi *delay end-to-end*, *packet loss*, dan *throughput* melalui teoritis dan pengamatan.
- 3) Membandingkan nilai perhitungan dan hasil pengukuran dengan standar yang dikeluarkan oleh ITU-T G 1010.

#### 4.2 Perancangan dan Instalasi

Perancangan sistem yang dilakukan adalah penyusunan perangkat yang disesuaikan dengan jaringan WLAN yang dibutuhkan. Instalasi sistem terdiri beberapa tahapan, meliputi instalasi perangkat lunak WireShark, dan VLC Media Player, serta *wireless router*. Gambar 4.1. merupakan gambar konfigurasi jaringan



Gambar 4.1 Konfigurasi Jaringan  
Sumber : Perancangan

Seperti yang sudah di jelaskan dalam bab 3 jaringan terdiri atas 1 buah laptop sebagai server dan 5 buah laptop sebagai client dan 1 buah wireless router. Untuk konfigurasi ip ditunjukkan oleh tabel 4.1

Tabel 4.1 Konfigurasi IP

| Alamat IP    | Keterangan             |
|--------------|------------------------|
| 192.168.1.1  | <i>Wireless Router</i> |
| 192.168.1.19 | <i>Pc server</i>       |
| 192.168.1.28 | <i>Pc client 1</i>     |
| 192.168.1.31 | <i>Pc client 2</i>     |
| 192.168.1.33 | <i>Pc client 3</i>     |
| 192.168.1.38 | <i>Pc client 4</i>     |
| 102.168.1.36 | <i>Pc client 5</i>     |

Sumber : Perancangan

Adapun fungsi-fungsi perangkat keras yang digunakan dalam jaringan WLAN adalah sebagai berikut:

Tabel 4.2 Fungsi Perangkat Keras

| No | Perangkat Keras          | Kegunaan                                                           |
|----|--------------------------|--------------------------------------------------------------------|
| 1  | Media Server (PC Server) | Penyedia layanan <i>Live Streaming</i>                             |
| 2  | <i>Wireless Router</i>   | Pengatur jalannya data dan proses <i>switching</i>                 |
| 3  | Laptop (User)            | Menerima <i>filestreaming</i> dan menjalankan <i>filestreaming</i> |

Sumber : Perancangan

Penelitian ini menggunakan perangkat-perangkat yang memiliki spesifikasi tersendiri, berikut adalah spesifikasi perangkat yang digunakan,

a. Wireless Router

*Wireless Router* yang digunakan adalah jenis TP-LINK TL-WR74IND. Router merupakan perangkat *switching* yang digunakan dalam penelitian ini yang ditunjukkan oleh gambar 4.2



Gambar 4.2 Wireless Router TP-LINK TL-WR741ND  
(Sumber: <http://www.tp-link.com/en/products/details/TL-WR741ND.html>)

Berikut spesifikasi dari Wireless Router yang digunakan ditunjukkan pada Tabel 4.3.

Tabel 4.3 Spesifikasi Wireless Router

| Spesifikasi     | Keterangan                               |
|-----------------|------------------------------------------|
| Standards       | IEEE 802.11n, IEEE 802.11g, IEEE 802.11b |
| Frequency Range | 2.4-2.4835GHz                            |
| Antenna Type    | 5dBi Detachable Omni Directional Antenna |
| Protocols       | Supports IPv4 and IPv6                   |
| Dimensions      | 6.9 x 4.6 x 1.3 in. (174 x 118 x 33 mm)  |

(Sumber: <http://www.tp-link.com/en/products/details/TL-WR741ND.html>)

#### b. PC Server

PC server digunakan penyedia layanan *Live Streaming*. Dalam penelitian ini menggunakan pc ASUS X550. Berikut spesifikasi ASUS X550 ditunjukkan pada Tabel 4.4:

Tabel 4.4 Spesifikasi Pc Server

| Spesifikasi    | Keterangan                                              |
|----------------|---------------------------------------------------------|
| Layar          | 15.6 Inch Resolusi HD 1366×768 Rasio 16:9               |
| Chipset        | AMD A76M FCH                                            |
| Processor      | AMD APU A10-5750M                                       |
| VGA            | AMD Radeon HD 8650G + AMD Radeon HD 8670M (Dual Grafis) |
| Penyimpanan    | 500HDD GB                                               |
| Sistem Operasi | Windows 10                                              |

(Sumber: <http://www.tabloidlaptop.com/2015/11/asus-x550dp.html>)

#### c. PC Client

PC client digunakan sebagai sarana meminta layanan *live streaming* dan aplikasi yang digunakan adalah VLC Media player yang diinstal pada PC client. Pada tabel 4.5 menunjukkan tipe laptop client.

Tabel 4.5 Tipe Laptop Client

| Client | Tipe Laptop           |
|--------|-----------------------|
| 1      | ASUS A46CB            |
| 2      | HP 4B11AU             |
| 3      | LENOVO YP410          |
| 4      | ASUS KD5D             |
| 5      | TOSHIBA SATELITE L745 |

Sumber : Perancangan

### 4.3 Langkah-Langkah Proses *Streaming* pada *Server*

Proses streaming video menggunakan laptop ASUS X 550 yang sudah diinstallasi VLC. Berikut penjelasan masing masing bagan untuk proses streaming:

1. Pastikan Laptop *server* dan *client* terhubung dengan jaringan WLAN.

Gambar 4.4 Koneksi ke *Wireless Router* pada jaringan WLAN

Sumber : Perancangan

2. Lakukan perintah PING dari *command prompt windows*. Ketik PING 192.168.1.38. Kemudian tunggu *reply* dari laptop *client*.

```
C:\Users\Fatur>ping 192.168.1.38

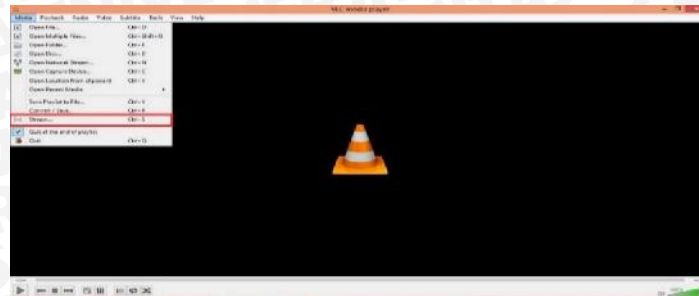
Pinging 192.168.1.38 with 32 bytes of data:
Reply from 192.168.1.22: Destination host unreachable.
Reply from 192.168.1.38: bytes=32 time=2442ms TTL=64
Reply from 192.168.1.38: bytes=32 time=8ms TTL=64
Reply from 192.168.1.38: bytes=32 time=1015ms TTL=64

Ping statistics for 192.168.1.38:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 8ms, Maximum = 2442ms, Average = 1155ms
```

Gambar 4.5 PING menuju Destinasi

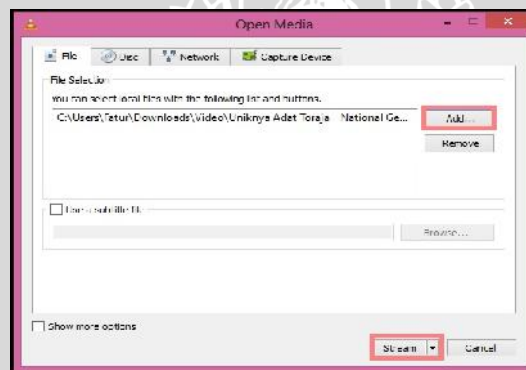
Sumber : Perancangan

3. Buka Aplikasi VLC lalu dan pilih menu “Media” dan pilih submenu “stream” seperti pada gambar



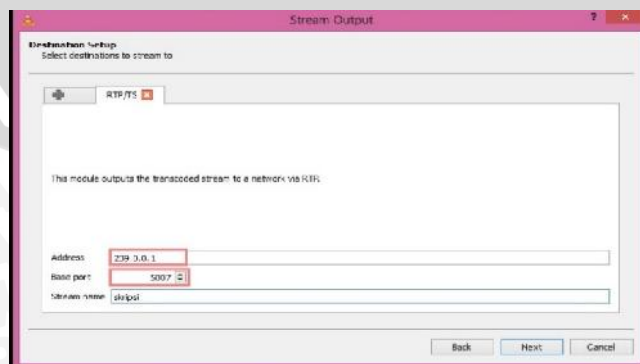
Gambar 4.6 Tampilan menu Media VLC  
Sumber : Perancangan

4. Memilih video yang akan di *stream*. Video yang di *stream* adalah Uniknya Adat Toraja - National Geographic - YouTube.MP4. Dengan Format MP4 dan durasi 3 menit 52 detik. Lalu pilih “Stream”.



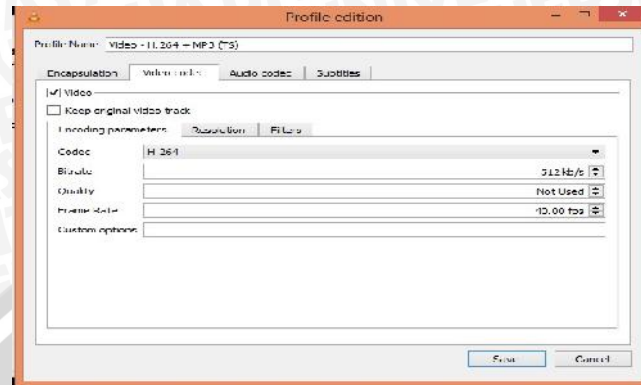
Gambar 4.7 Kotak dialog pemilihan video  
Sumber : Perancangan

5. Kotak dialog *destination setup*. Lalu pilih RTP kemudian klik “add”. Lalu mengisi ip dengan ip multicast yaitu : 239.0.0.1 dengan port 5007. Kemudian klik “next”



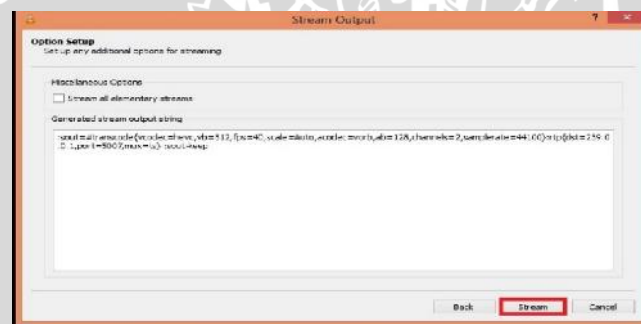
Gambar 4.8 Kotak dialog *destination setup*  
Sumber : Perancangan

6. Kotak dialog *Transcoding Options*. Dalam Kotak dialog ini dapat mengubah-ubah *bit rate* dan *frame rate* video. Kemudian pilih “*video codec*” lalu setelah memilih *frame rate* dan *bit rate* video. Lalu klik “*save*”.



Gambar 4.9 Kotak dialog *transcoding options*  
Sumber : Perancangan

7. Kotak dialog *stream output*. Lalu klik “*stream*” dan video sudah di stream.



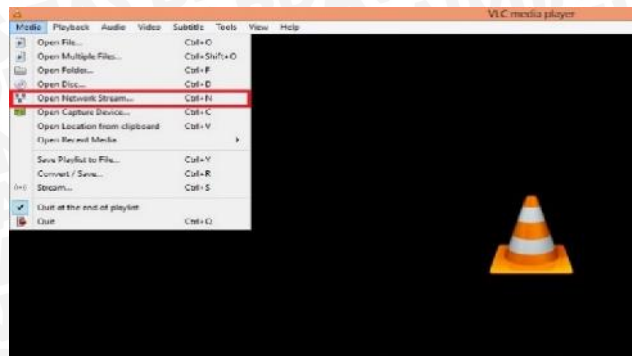
Gambar 4.10 Kotak Dialog *stream output*  
Sumber : Perancangan

#### 4.4. Langkah-Langkah Proses *Streaming* pada *Client*

*Streaming video* dilakukan pada laptop *client* yang sudah diinstalasi VLC Media Player. Instalasi dilakukan agar *client* dapat melakukan *streaming* dengan alamat video yang sudah dijalankan pada *server*.

Berikut merupakan langkah-langkah :

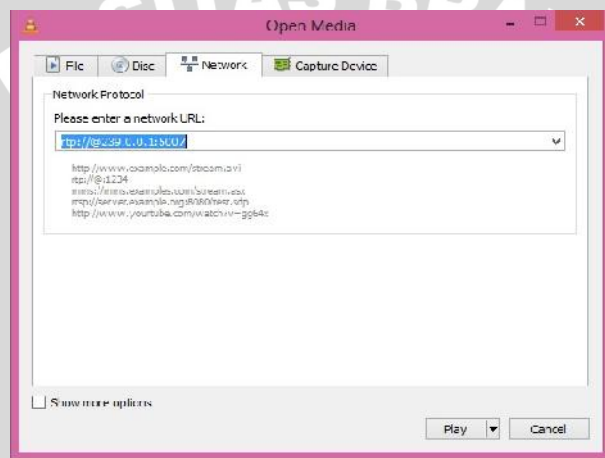
1. Buka perangkat lunak VLC Media Player. Kemudian pilih menu “Media” dan pilih submenu “Buka Stream Jaringan”



Gambar 4.12 Tampilan menu *media* VLC

Sumber : Perancangan

2. Isi alamat video streaming pada kotak dialog sesuai dengan alamat video pada server. Alamatnya : `rtp://@239.0.0.1:5007`.



Gambar 4.13 Kotak dialog pengisian alamat video *streaming*

Sumber : Perancangan

3. Klik “play” kemudian video berjalan.



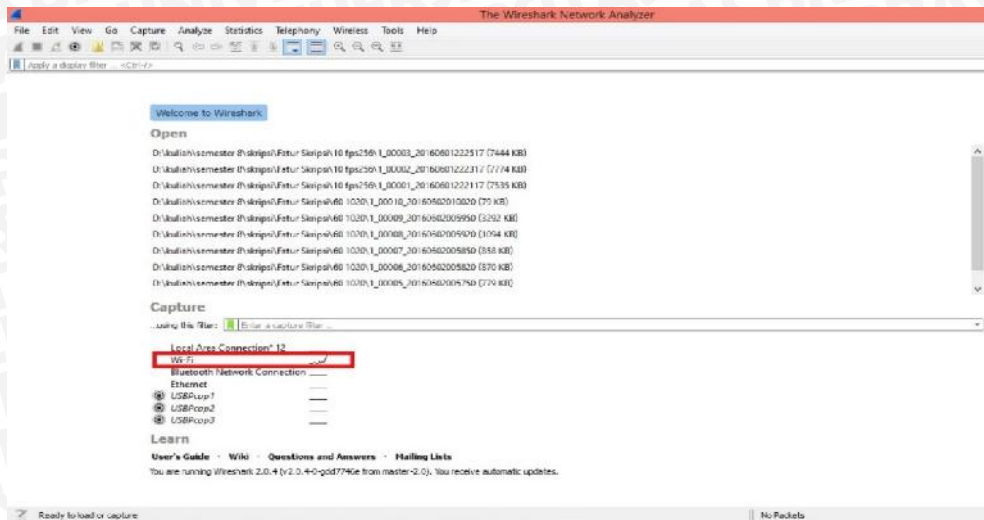
Gambar 4.14 Tampilan video *streaming*

Sumber : Perancangan

#### 4.5 Proses *Capturing* Data Menggunakan Wireshark

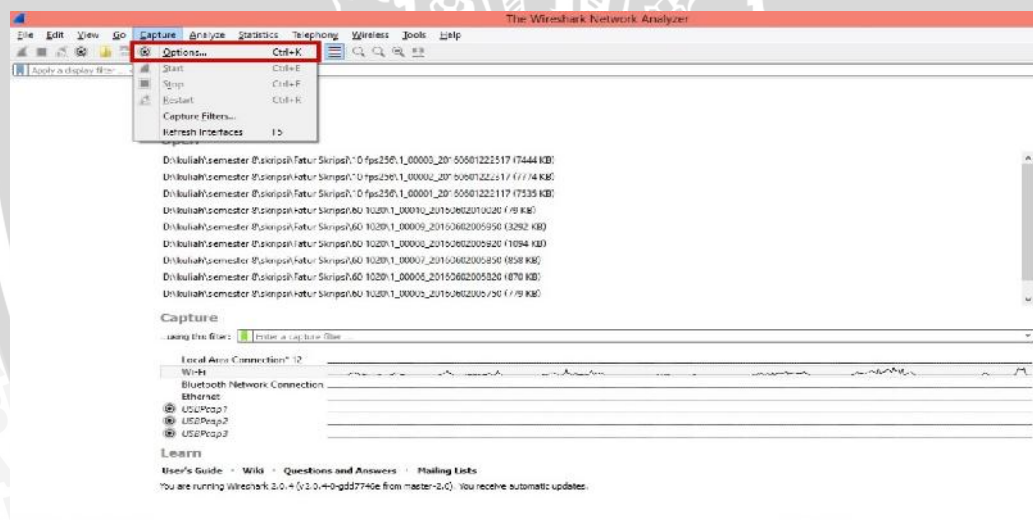
Proses *capturing* data pada *laptop client* dilakukan dengan *network analyzer* Wireshark. Proses *capturing* data dilakukan saat video *streaming* sedang diputar. Berikut adalah proses *capturing* data menggunakan perangkat lunak Wireshark

1. Buka perangkat lunak Wireshark dan pilih “*Interface List*” untuk memilih device interface yang digunakan. Klik pada pilihan “*Wi-Fi*”



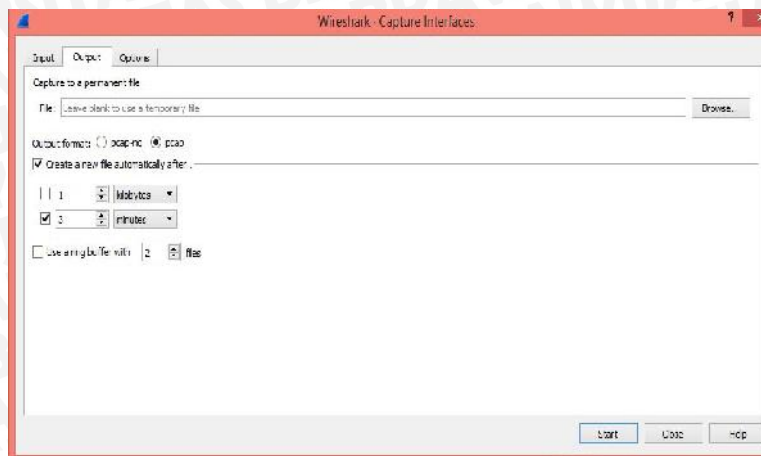
Gambar 4.15 Tampilan *Interface List* Wireshark  
Sumber : Perancangan

2. Pilih menu “*Options*” untuk memilih waktu *capture* dan memulai proses *capturing* data.



Gambar 4.16 Tampilan *Capture* menu  
Sumber : Perancangan

3. Berikan tanda ✓ pada pilihan *1 minute* dan klik “*start*”



Gambar 4.17 Tampilan pada Capture Interface  
Sumber : Perancangan

#### 4. Tampilan proses *capturing* wireshark

| No. | Time     | Source        | Destination | Protocol | Length | Time delta from previous captured frame | Time delta from previous displayed frame | Info                  |
|-----|----------|---------------|-------------|----------|--------|-----------------------------------------|------------------------------------------|-----------------------|
| 1   | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.000000000                             | 0.000000000                              | 54616 → 5667 Len=1328 |
| 2   | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.000001000                             | 0.000001000                              | 54616 → 5667 Len=1328 |
| 3   | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.000201000                             | 0.000201000                              | 54616 → 5667 Len=1328 |
| 4   | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.000401000                             | 0.000401000                              | 54616 → 5667 Len=1328 |
| 5   | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.000601000                             | 0.000601000                              | 54616 → 5667 Len=1328 |
| 6   | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.000801000                             | 0.000801000                              | 54616 → 5667 Len=1328 |
| 7   | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.001001000                             | 0.001001000                              | 54616 → 5667 Len=1328 |
| 8   | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.001201000                             | 0.001201000                              | 54616 → 5667 Len=1328 |
| 9   | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.001401000                             | 0.001401000                              | 54616 → 5667 Len=1328 |
| 10  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.001601000                             | 0.001601000                              | 54616 → 5667 Len=1328 |
| 11  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.001801000                             | 0.001801000                              | 54616 → 5667 Len=1328 |
| 12  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.002001000                             | 0.002001000                              | 54616 → 5667 Len=1328 |
| 13  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.002201000                             | 0.002201000                              | 54616 → 5667 Len=1328 |
| 14  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.002401000                             | 0.002401000                              | 54616 → 5667 Len=1328 |
| 15  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.002601000                             | 0.002601000                              | 54616 → 5667 Len=1328 |
| 16  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.002801000                             | 0.002801000                              | 54616 → 5667 Len=1328 |
| 17  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.003001000                             | 0.003001000                              | 54616 → 5667 Len=1328 |
| 18  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.003201000                             | 0.003201000                              | 54616 → 5667 Len=1328 |
| 19  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.003401000                             | 0.003401000                              | 54616 → 5667 Len=1328 |
| 20  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.003601000                             | 0.003601000                              | 54616 → 5667 Len=1328 |
| 21  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.003801000                             | 0.003801000                              | 54616 → 5667 Len=1328 |
| 22  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.004001000                             | 0.004001000                              | 54616 → 5667 Len=1328 |
| 23  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.004201000                             | 0.004201000                              | 54616 → 5667 Len=1328 |
| 24  | 0.000000 | 192.168.0.107 | 239.0.0.1   | UDP      | 1376   | 0.004401000                             | 0.004401000                              | 54616 → 5667 Len=1328 |

Frame 1: 1376 bytes on wire (10960 bits), 1376 bytes captured (10960 bits) on interface 0  
 Ethernet II, Src: Azurewax.al:9b:5f (51:27:1e:a1:9b:5f), Dst: LiteonTe.60:23:7c (28:1e:d8:60:23:7c)  
 Internet Protocol Version 4, Src: 192.168.0.107, Dst: 239.0.0.1  
 User Datagram Protocol, Src Port: 54616, Dst Port: 5667

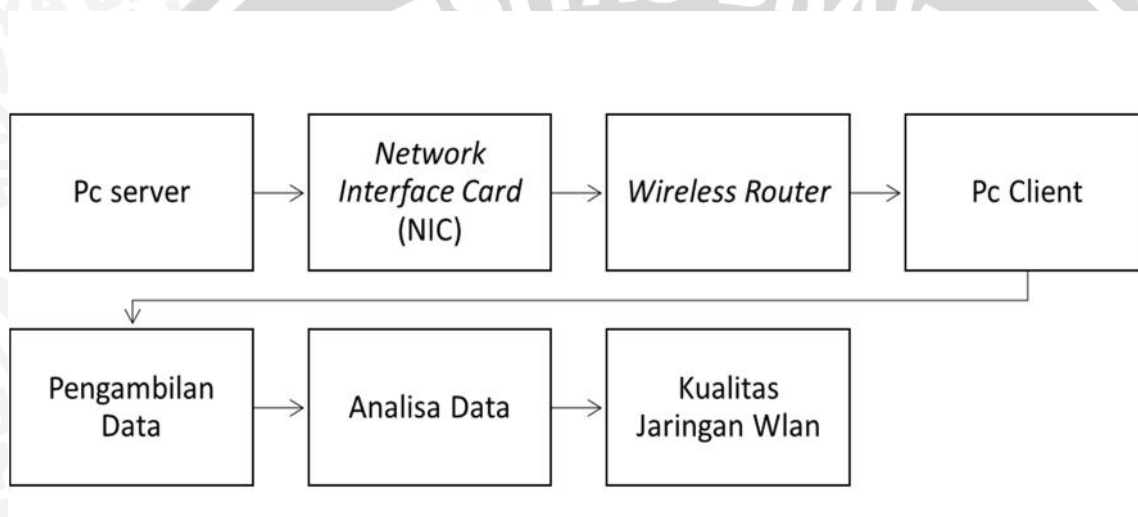
Gambar 4.18 Tampilan *Capturing* data Wireshark  
Sumber : Perancangan

#### 4.6 Hasil Dan Pembahasan

Pada pengambilan data ini akan menampilkan hasil eksperimen penggunaan layanan *live streaming* dengan memberikan beberapa variasi nilai *frame rate* dan *bit rate* video serta menampilkan nilai dari masing-masing parameter antara lain *delay*, *throughput*, dan *packet loss* berdasarkan hasil pengamatan menggunakan *network analyzer* Wireshark. Pengambilan data dilakukan pada tanggal 1 Juni 2016 jam 16.00 hingga jam 23.35 bertempat di laboratorium saluran transmisi dan gelombang mikro. Pengambilan data dilakukan 10 percobaan untuk tiap-tiap parameter. Sedangkan untuk perhitungan teoritis dilakukan dengan rumus yang sesuai dengan *quality of service* dari jaringan *wireless local area network* (WLAN).

#### 4.6.1 Hasil Melalui Eksperimen

Dari hasil eksperimen data diperoleh menggunakan perangkat lunak yaitu wireshark. Data yang diperoleh meliputi *Throughput*, *packet loss ratio*, dan *delay end-to-end*. Dalam eksperimen ini menggunakan 5 buah laptop sebagai *client*. Dan 1 buah laptop sebagai *server*. Proses *streaming* dijalankan dengan beberapa tahap, tahapan-tahapan yang berlangsung ialah 2 *client* melakukan proses *streaming* dalam waktu yang bersamaan. Sementara selang waktu 1 menit 2 *client* melakukan proses *streaming*, dan 1 *client* melakukan proses *streaming* setelah selang waktu 2 menit. Proses pengambilan data dilakukan 10 kali untuk tiap-tiap parameter pengamatan. Adapun blok diagram untuk pengujian ini ditunjukkan pada gambar 4.19



Gambar 4.19 Blok diagram pengujian  
Sumber : Perancangan

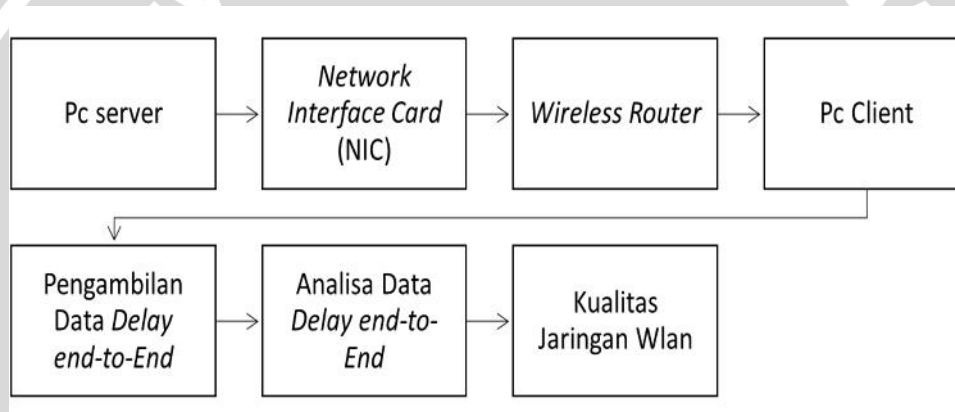
Berikut merupakan penjelasan blok diagram dari pengujian :

1. Pc Server mengirimkan file video streaming.
2. Mengatur jalannya pengiriman data dan menerima aliran data.
3. Wireless Router melakukan pengalamatan. Serta meneruskan data menuju pc client
4. Client Melakukan permintaan bergabung dengan grup multicast. Setelah Bergabung maka wireless router akan meneruskan file video streaming dari pc client.
5. Pengambilan data dilakukan 10x dengan untuk tiap tiap parameter.
6. Analisis data untuk tiap tiap parameter quality of service.
7. Membandingkan data hasil pengamatan dengan rekomendasi ITU-T G.1010 dan ITU-T G.114

Pengambilan data dilakukan dengan pengiriman file video streaming dengan menggunakan aplikasi VLC yang sudah dikonfigurasi, Parameter yang dicatat adalah waktu pengiriman, jumlah dan besar paket yang diterima, dan transfer rate dari proses pengiriman (*throughput*). Parameter ini diamati dengan bantuan aplikasi WIRESHARK. Pada penelitian ini dilakukan pengamatan dan analisa terhadap QoS dari Video *streaming* dengan menggunakan pengaturan *Frame Rate* dengan mengubah nilainya yaitu mulai dari 10 fps, 25 fps, 40 fps, 60 fps. Serta mengubah *bitrate*, mulai dari 256 Kbps, 512 Kbps, 1024 Kbps.

#### 4.6.1.1 Delay End-to-End

*Delay end-to-end* pada jaringan merupakan penjumlahan *delay-delay* yang ada dalam perjalanan paket dari sumber ke tujuannya pada setiap hop. Berikut merupakan blok diagram pengujian untuk mendapatkan nilai *delay end-to-end* yang ditunjukkan oleh gambar 4.20



Gambar 4.20 Blok diagram pengujian *delay end-to-end*  
Sumber : Perancangan

Berikut merupakan penjelasan blok diagram dari pengujian :

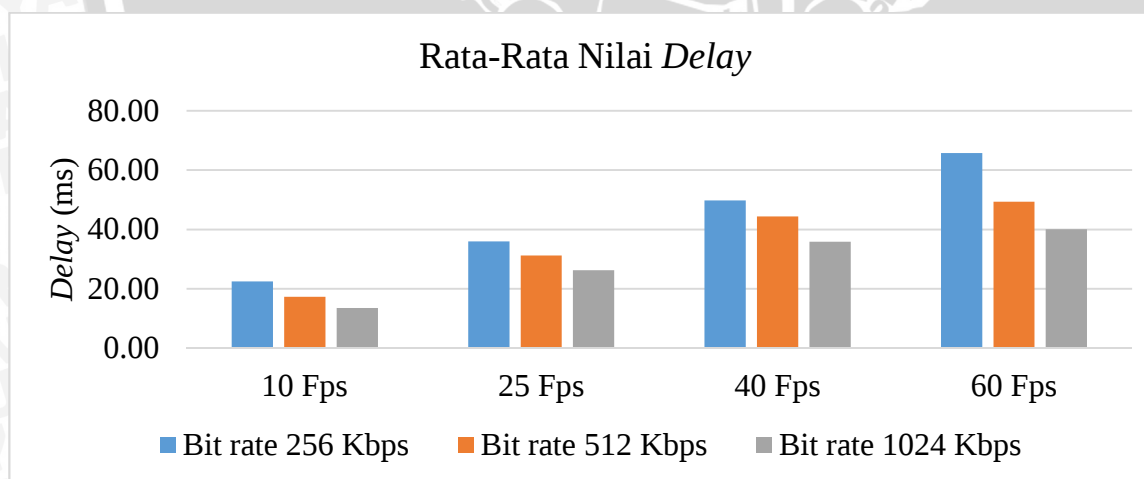
1. Pc Server mengirimkan file video streaming.
2. Mengatur jalannya pengiriman data dan menerima aliran data.
3. Wireless Router melakukan pengalamatan. Serta meneruskan data menuju pc client
4. Client Melakukan permintaan bergabung dengan grup multicast. Setelah Bergabung maka wireless router akan meneruskan file video streaming dari pc client.
5. Pengambilan data *Delay end-to-end* dilakukan 10x dengan untuk tiap tiap parameter.
6. Analisis data untuk *Delay end-to-end*.
7. Membandingkan data hasil pengamatan dengan rekomendasi ITU-T G.114

Pengambilan data *delay* pada Wireshark menggunakan persamaan 2-3. Hasil *capturing* wireshark terhadap *delay* sistem *live streaming* ditunjukkan oleh Tabel 4.6.

Tabel 4.6 Nilai *delay end-to-end* hasil *capturing* wireshark

| Frame Rate (Fps) | Bit Rate (Kbps) | Nilai Delay end-to-end (ms) |       |       |       |       |       |       |       |       |       | Rata-Rata (ms) |
|------------------|-----------------|-----------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|----------------|
|                  |                 | Percobaan ke-               |       |       |       |       |       |       |       |       |       |                |
|                  |                 | 1                           | 2     | 3     | 4     | 5     | 6     | 7     | 8     | 9     | 10    |                |
| 10               | 256             | 21,23                       | 20,65 | 21,22 | 21,47 | 24,32 | 25,54 | 22,12 | 25,65 | 20,20 | 21,84 | 22,43          |
|                  | 512             | 16,60                       | 17,03 | 16,23 | 17,57 | 18,69 | 15,09 | 18,49 | 16,12 | 17,25 | 19,26 | 17,23          |
|                  | 1024            | 16,81                       | 13,90 | 14,40 | 11,96 | 12,42 | 13,82 | 12,77 | 10,43 | 12,99 | 16,01 | 13,55          |
| 25               | 256             | 42,60                       | 26,88 | 31,24 | 31,76 | 34,60 | 31,47 | 45,51 | 47,77 | 41,95 | 25,94 | 35,97          |
|                  | 512             | 24,79                       | 41,25 | 23,79 | 31,71 | 28,12 | 31,95 | 38,53 | 29,53 | 29,07 | 32,84 | 31,16          |
|                  | 1024            | 12,01                       | 52,63 | 9,78  | 54,77 | 13,58 | 65,88 | 9,30  | 7,32  | 18,73 | 18,70 | 26,27          |
| 40               | 256             | 35,66                       | 34,84 | 39,86 | 51,43 | 53,09 | 45,14 | 50,32 | 47,86 | 75,72 | 63,58 | 49,75          |
|                  | 512             | 34,59                       | 46,61 | 54,82 | 33,00 | 42,67 | 44,55 | 45,05 | 43,45 | 52,28 | 46,72 | 44,37          |
|                  | 1024            | 35,31                       | 30,29 | 45,54 | 49,87 | 32,01 | 35,48 | 37,85 | 34,19 | 29,66 | 28,71 | 35,89          |
| 60               | 256             | 43,68                       | 51,76 | 62,02 | 53,70 | 49,31 | 79,19 | 79,31 | 59,94 | 96,73 | 81,45 | 65,71          |
|                  | 512             | 51,25                       | 51,41 | 34,24 | 66,96 | 41,25 | 42,69 | 58,19 | 41,75 | 46,82 | 59,22 | 49,38          |
|                  | 1024            | 55,02                       | 41,12 | 30,30 | 29,32 | 44,02 | 37,87 | 43,03 | 31,08 | 39,51 | 48,92 | 40,02          |

Kemudian jika rata-rata nilai *delay end-to-end* direpresentasikan ke dalam bentuk grafik hubungan antara *bit rate* dan *frame rate* dengan *delay* yang diterima ditunjukkan pada Gambar 4.21



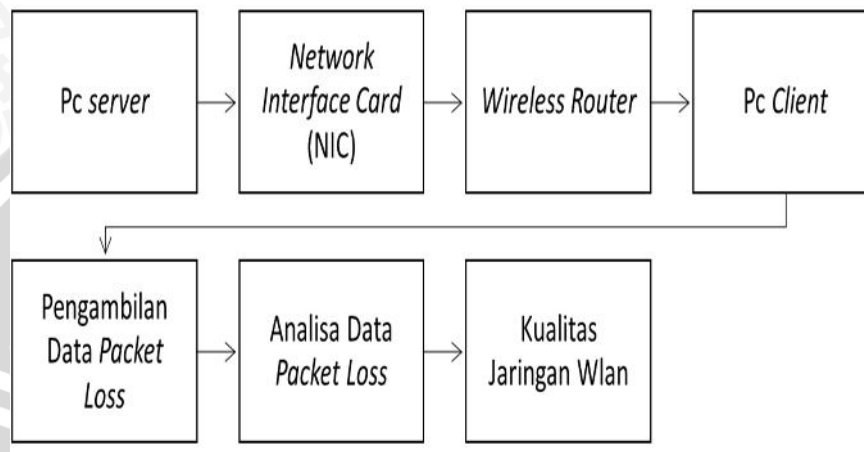
Gambar 4.21 Grafik hubungan antara *Bit rate* dan *Frame rate* video dengan *Delay end-to-end* hasil *capturing* wireshark

Berdasarkan hasil *capturing* wireshark menunjukkan nilai *delay* yang berubah-ubah sesuai dengan *frame rate* dan *bit rate* video, semakin besar nilai *bit rate* maka semakin kecil nilai *delay* yang didapatkan, sementara semakin besar *frame rate* maka semakin besar pula

delay. Berdasarkan hasil nilai *delay end-to-end* layanan *live streaming* memiliki kualitas yang baik karena delay < 150ms sesuai dengan standart ITU-T G.1010.

#### 4.6.1.2 Packet Loss

*Packet loss* dapat diartikan sebagai hilangnya sejumlah paket data pada jaringan komputer selama proses transmisi paket data. Berikut merupakan blok diagram pengujian untuk mendapatkan nilai *packet loss* yang ditunjukkan oleh gambar 4.22.



Gambar 4.22 Blok diagram pengujian *Packet loss*  
Sumber : Perancangan

Berikut merupakan penjelasan blok diagram dari pengujian :

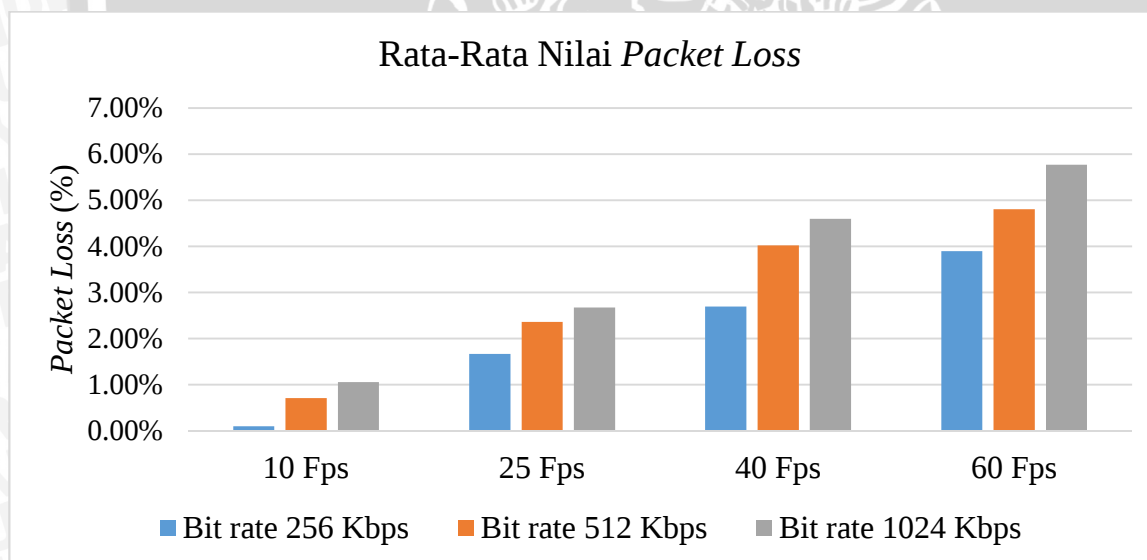
1. Pc Server mengirimkan file video streaming.
2. Mengatur jalannya pengiriman data dan menerima aliran data.
3. Wireless Router melakukan pengalamatan. Serta meneruskan data menuju pc client
4. Client Melakukan permintaan bergabung dengan grup multicast. Setelah Bergabung maka wireless router akan meneruskan file video streaming dari pc client.
5. Pengambilan data *packet loss* dilakukan 10x dengan untuk tiap tiap parameter.
6. Analisis data untuk *Packet loss*.
7. Membandingkan data hasil pengamatan dengan rekomendasi ITU-T G.1010

Perhitungan nilai persentase *packet loss* menggunakan persamaan 2.14 dimana membutuhkan data primer seperti jumlah paket yang diterima dan jumlah paket yang hilang, kedua data primer tersebut didapatkan dari hasil *capturing* trafik menggunakan Wireshark seperti yang diitunjukkan pada Tabel 4.7

Tabel 4.7 Nilai Packet Loss

| Frame Rate (Fps) | Bit Rate (Kbps) | Nilai Packet loss (%) |       |       |       |       |       |       |       |       |       | Rata-Rata |
|------------------|-----------------|-----------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-----------|
|                  |                 | Percobaan Ke-         |       |       |       |       |       |       |       |       |       |           |
|                  |                 | 1                     | 2     | 3     | 4     | 5     | 6     | 7     | 8     | 9     | 10    |           |
| 10               | 256             | 0,00%                 | 0,72% | 0,00% | 0,00% | 0,05% | 0,02% | 0,01% | 0,06% | 0,03% | 0,12% | 0,10%     |
|                  | 512             | 0,80%                 | 0,30% | 0,40% | 0,25% | 0,55% | 0,18% | 0,42% | 1,96% | 1,18% | 1,05% | 0,71%     |
|                  | 1024            | 1,13%                 | 0,84% | 0,69% | 1,27% | 1,65% | 0,88% | 1,33% | 0,89% | 0,81% | 1,04% | 1,05%     |
| 25               | 256             | 1,69%                 | 1,63% | 1,83% | 1,71% | 2,05% | 1,28% | 1,55% | 1,93% | 1,58% | 1,44% | 1,67%     |
|                  | 512             | 1,34%                 | 2,46% | 1,78% | 1,59% | 3,08% | 2,98% | 2,39% | 2,76% | 2,59% | 2,64% | 2,36%     |
|                  | 1024            | 1,93%                 | 2,63% | 2,49% | 3,78% | 2,68% | 1,40% | 2,84% | 2,45% | 3,73% | 2,80% | 2,67%     |
| 40               | 256             | 1,26%                 | 4,10% | 1,90% | 3,11% | 2,09% | 1,60% | 3,20% | 4,70% | 1,52% | 3,43% | 2,69%     |
|                  | 512             | 3,60%                 | 6,79% | 4,20% | 5,98% | 4,87% | 2,79% | 2,40% | 3,83% | 2,20% | 3,53% | 4,02%     |
|                  | 1024            | 4,33%                 | 3,74% | 3,76% | 5,58% | 4,20% | 2,12% | 7,30% | 4,70% | 4,24% | 6,00% | 4,60%     |
| 60               | 256             | 5,02%                 | 3,27% | 4,57% | 4,97% | 2,13% | 4,17% | 5,67% | 3,70% | 2,89% | 2,56% | 3,89%     |
|                  | 512             | 5,20%                 | 3,80% | 6,40% | 4,80% | 2,91% | 4,66% | 4,73% | 6,15% | 5,78% | 3,59% | 4,80%     |
|                  | 1024            | 5,42%                 | 4,10% | 7,42% | 5,80% | 6,30% | 3,15% | 6,80% | 8,10% | 4,78% | 5,81% | 5,77%     |

Kemudian jika rata-rata nilai *packet loss* direpresentasikan ke dalam bentuk grafik hubungan antara *bit rate* dan *frame rate* dengan *packet loss* yang diterima ditunjukkan pada Gambar 4.23



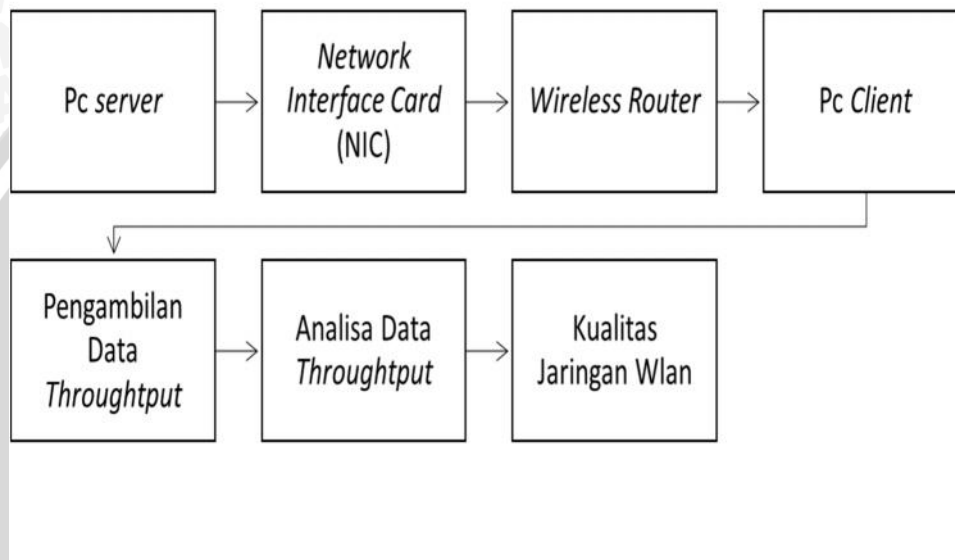
Gambar 4.23 Grafik hubungan antara *Bit rate* dan *Frame rate* video dengan *packet loss* hasil *capturing* wireshark

Berdasarkan hasil *capturing* wireshark nilai *packet loss* terbesar pada video 60 fps dengan bitrate 1024 Kbps, sementara untuk nilai *packet loss* terkecil pada video 10 fps dengan bitrate 256 Kbps. Dari data tersebut menunjukkan apabila semakin tinggi nilai *bitrate* dan *frame rate* video maka nilai *packet loss* semakin besar. Berdasarkan ITU-T-G1010 layanan live streaming bagus tetapi untuk video dengan frame rate 60 bit rate 1024 Kbps

layanan menurun dengan nilai packet loss sebesar 5,77% tetapi masih bisa dikategorikan cukup yaitu dengan standar 5%-10%.

#### 4.6.1.3 Throughput

*Throughput* adalah jumlah rata-rata paket yang sukses diterima atau dikirimkan oleh saluran penerima atau pemancar per detik. *Throughput* memiliki satuan *byte/detik*. Berikut merupakan blok diagram pengujian untuk mendapatkan nilai *throughput* yang ditunjukkan oleh gambar 4.24



Gambar 4.24 Blok diagram pengujian *throughput*

Sumber : Perancangan

Berikut merupakan penjelasan blok diagram dari pengujian :

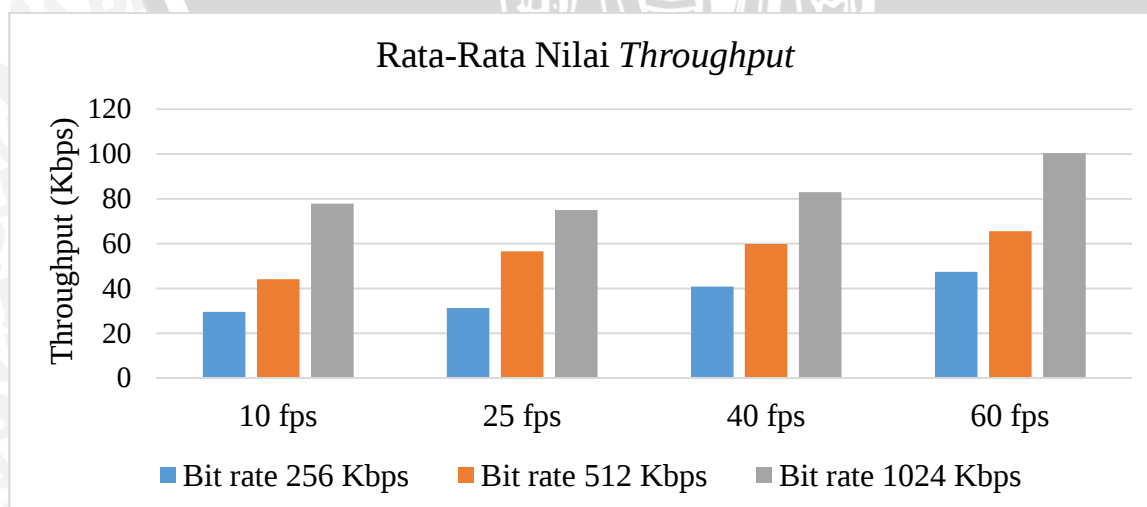
1. Pc Server mengirimkan file video streaming.
2. Mengatur jalannya pengiriman data dan menerima aliran data.
3. Wireless Router melakukan pengalamatan. Serta meneruskan data menuju pc client
4. Client Melakukan permintaan bergabung dengan grup multicast. Setelah Bergabung maka wireless router akan meneruskan file video streaming dari pc client.
5. Pengambilan data *Throughput* dilakukan 10x dengan untuk tiap tiap parameter.
6. Analisis data untuk *Throughput*.
7. Dengan melihat nilai *throughput* dengan nilai *throughput* yang semakin tinggi maka jaringan baik.

Berikut merupakan hasil *capturing* wireshark untuk nilai *throughput* tiap tiap parameter yang ditunjukkan oleh tabel 4.8

Tabel 4.8 Nilai *Throughput* hasil *capturing* wireshark

| Frame Rate (Fps) | Bit Rate (Kbps) | Nilai Throughput (Kbps) |    |    |     |    |    |     |     |     |    | Rata-Rata (Kbps) |
|------------------|-----------------|-------------------------|----|----|-----|----|----|-----|-----|-----|----|------------------|
|                  |                 | Percobaan ke-           |    |    |     |    |    |     |     |     |    |                  |
|                  |                 | 1                       | 2  | 3  | 4   | 5  | 6  | 7   | 8   | 9   | 10 |                  |
| 10               | 256             | 29                      | 32 | 27 | 24  | 38 | 34 | 22  | 26  | 31  | 33 | 29,6             |
|                  | 512             | 36                      | 43 | 39 | 54  | 41 | 47 | 44  | 49  | 43  | 46 | 44,2             |
|                  | 1024            | 56                      | 84 | 88 | 75  | 66 | 79 | 68  | 69  | 101 | 92 | 77,8             |
| 25               | 256             | 37                      | 36 | 33 | 25  | 23 | 26 | 26  | 28  | 35  | 44 | 31,3             |
|                  | 512             | 58                      | 48 | 79 | 44  | 62 | 53 | 56  | 66  | 53  | 47 | 56,6             |
|                  | 1024            | 58                      | 62 | 71 | 63  | 67 | 83 | 86  | 117 | 72  | 71 | 75               |
| 40               | 256             | 41                      | 43 | 48 | 34  | 27 | 42 | 35  | 56  | 45  | 38 | 40,9             |
|                  | 512             | 63                      | 65 | 63 | 62  | 54 | 51 | 61  | 52  | 66  | 61 | 59,8             |
|                  | 1024            | 80                      | 79 | 83 | 92  | 75 | 80 | 82  | 83  | 78  | 98 | 83               |
| 60               | 256             | 69                      | 47 | 43 | 40  | 37 | 42 | 46  | 39  | 61  | 51 | 47,5             |
|                  | 512             | 84                      | 68 | 75 | 68  | 49 | 58 | 56  | 72  | 57  | 69 | 65,6             |
|                  | 1024            | 80                      | 98 | 94 | 103 | 99 | 97 | 104 | 130 | 104 | 95 | 100,4            |

Kemudian jika rata-rata nilai *throughput* Tabel 4.8 direpresentasikan ke dalam bentuk grafik hubungan antara *frame rate* dan *bit rate* video dengan *throughput* hasil *capturing* wireshark ditunjukkan oleh Gambar 4.25



Gambar 4.25 Grafik hubungan antara *frame rate* dan *bit rate* video dengan *throughput* hasil *capturing* wireshark

Berdasarkan data pada tabel 4.8 nilai *throughput* terbesar pada video 60 fps dengan bitrate 1024 Kbps, sementara untuk nilai *throughput* terkecil pada video 10 fps dengan bitrate 256 Kbps. Dari data tersebut menunjukkan apabila semakin tinggi nilai *bitrate* dan *frame rate* video maka nilai *throughput* semakin besar.

#### 4.6.2 Hasil Perhitungan Teoritis

Pada bagian ini dihasilkan parameter jaringan yaitu *delay end-to-end*, *packet loss* dan *throughput* melalui perhitungan dengan literatur yang ada, standar perangkat atau standar yang telah ditetapkan.

##### 4.6.2.1 Spesifikasi Paket Video Streaming

Spesifikasi paket video *streaming* yang digunakan adalah sebagai berikut :  
Pada penelitian ini menggunakan 2 jenis *codec*, yaitu *codec* H.264 untuk video dan *codec* AAC untuk audio, spesifikasi masing-masing *codec* dijelaskan dibawah:

- *Audio codec* adalah AAC  
Menggunakan *bandwidth* 126 Kbps, *delay codec* 54,6 ms dan *payload* maksimum 63 byte
- *Video codec* adalah H.264/MPEG-4 AVC  
Menggunakan *bandwidth* 10-548 Kbps, *delay codec* 50 ms dan *payload* maksimum 254 byte

##### 4.6.2.2 Delay End-to-End

Pada perhitungan *delay end-to-end* secara teoritis meliputi *delay* CODEC, *delay* proses, *delay* transmisi, *delay* propagasi, *delay* antrian.

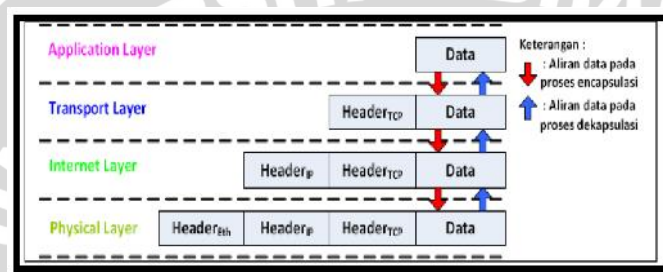
##### 4.6.2.2.1 Delay CODEC

*Delay* CODEC adalah *delay* yang disebabkan oleh pengkodean dari CODEC yang digunakan. *Delay* CODEC terdiri atas *delay* CODEC audio ( $t_a$ ) dan video ( $t_v$ ) masing-masing CODEC memiliki besaran *delay* audio dan *delay* video yang berbeda-beda. CODEC yang digunakan untuk audio adalah ACC memiliki *delay* 54,6 ms. CODEC yang digunakan pada video adalah H.262/MPEG-4 AVC yang memiliki *delay* sebesar 50 ms. Sehingga *Delay* Codec dapat dihitung dengan persamaan (2.4) :

$$\begin{aligned} t_{\text{codec}} &= t_a + t_v \\ &= 54,6 + 50 \\ &= 104,6 \text{ ms} \end{aligned}$$

#### 4.6.2.2.2 Delay Proses

*Delay* proses meliputi *delay* enkapsulasi dan *delay* deenkapsulasi. *Delay* enkapsulasi adalah waktu yang dibutuhkan untuk proses pemaketan data dan penambahan header sebelum dikirim ke tujuan. Enkapsulasi terjadi ketika sebuah protokol yang berada pada lapisan yang lebih rendah menerima data dari protokol yang berada pada lapisan yang lebih tinggi dan meletakkan data ke format data yang dipahami oleh protokol tersebut. Sedangkan *delay* deenkapsulasi adalah kebalikan dari proses enkapsulasi, sehingga data dapat terbaca di tujuan. Kedua *delay* ini dipengaruhi oleh panjang data, panjang header protokol yang digunakan, lapisan yang dilewati dan kecepatan pemrosesan data. Gambar 4.26 menggambarkan proses *delay* ini.



Gambar 4.26 Proses *Delay Enkapsulasi* dan *Dekapsulasi*  
 (Sumber: <http://www.citap.com/document/tcp-ip/tcpip011.html>)

Berdasarkan data wireshark dan studi literatur maka di dapat :

- Panjang header IPv4 ( $L_{Header\ Ipv4}$ ) = 20 byte/paket
- Panjang header NALU ( $L_{Header\ NALU}$ ) = 1 byte/paket
- Panjang header UDP ( $L_{Header\ UDP}$ ) = 3 byte/paket
- Panjang header RTP ( $L_{Header\ RTP}$ ) = 12 byte/paket
- Panjang header Ethernet ( $L_{Header\ Ethernet}$ ) = 14 byte/paket
- Panjang header CRC ( $L_{Header\ CRC}$ ) = 4 byte/paket
- Panjang header VLAN ( $L_{Header\ VLAN}$ ) = 4 byte/paket

Perhitungan *delay* enkapsulasi adalah sebagai berikut,

- Jumlah paket yang dikirim (N) = 16.181 paket
- Average Packet Size = 1.344,5 byte  
= 10.756 bit
- Jumlah total data yang dikirim = 174.702.696 bit
- Waktu pengiriman total data = 240 s

Perhitungan kecepatan pemrosesan pada terminal pengirim adalah sebagai berikut,

$$C_{proses} = \frac{N_k}{T_k}$$

$$C_{proses} = \frac{174.702.696}{240}$$

$$C_{proses} = 90.990,9875 \text{ bps}$$

Panjang *header* ( $L'$ ), didapatkan

$$L' = L_{Header \text{ Ipv4}} + L_{Header \text{ NALU}} + L_{Header \text{ UDP}} + L_{Header \text{ RTP}} + L_{Header \text{ FCS}} + L_{Header \text{ VLAN}}$$

$$L' = (20 + 1 + 8 + 12 + 14 + 4 + 4) \times 8 \text{ bit}$$

$$L' = 504 \text{ bit}$$

Jadi, perhitungan *delay* enkapsulasi adalah sebagai berikut:

$$t_{enc} = \frac{L_{Header \text{ Ipv4}} + L_{Header \text{ NALU}} + L_{Header \text{ UDP}} + L_{Header \text{ RTP}} + L_{Header \text{ Ethernet}} + L_{Header \text{ FCS}} + L_{Header \text{ VLAN}}}{C_{proses}}$$

$$t_{enc} = \frac{(20 + 1 + 8 + 12 + 14 + 4 + 4) \times 8 \text{ bit}}{90.990,9875}$$

$$t_{enc} = 0,000553901 \text{ sekon}$$

$$t_{enc} = \mathbf{0,553901 \text{ ms}}$$

Perhitungan kecepatan pemrosesan pada terminal pengirim adalah sebagai berikut,

$$C_{proses} = \frac{N_k}{T_k}$$

$$C_{proses} = \frac{174.702.696}{240}$$

$$C_{proses} = 90.990,9875 \text{ bps}$$

Jadi, perhitungan *delay* dekapsulasi adalah sebagai berikut:

$$t_{enc} = \frac{L_{Header \text{ Ipv4}} + L_{Header \text{ NALU}} + L_{Header \text{ UDP}} + L_{Header \text{ RTP}} + L_{Header \text{ Ethernet}} + L_{Header \text{ FCS}} + L_{Header \text{ VLAN}}}{C_{proses}}$$

$$t_{enc} = \frac{(20 + 1 + 8 + 12 + 14 + 4 + 4) \times 8 \text{ bit}}{90.990,9875}$$

$$t_{enc} = 0,000553901 \text{ sekon}$$

$$t_{dec} = \mathbf{0,553901 \text{ ms}}$$

Maka *delay* proses diperoleh dari persamaan 2.5 sebagai berikut :

$$t_{proses} = t_{enc} + t_{dec}$$

$$t_{proses} = 0,553901 \text{ ms} + 0,553901 \text{ ms}$$

$$t_{proses} = \mathbf{1,107802 \text{ ms}}$$

Tabel 4.9 berikut merupakan hasil nilai *delay* proses pada masing-masing *Frame rate* dan *bit rate* video.

Tabel 4.9 Nilai *Delay* Proses

| <i>Frame Rate</i><br>(Fps) | <i>Bit Rate</i><br>(Kbps) | <i>Delay</i><br>proses<br>(ms) |
|----------------------------|---------------------------|--------------------------------|
| 10                         | 256                       | 1,8527097                      |
|                            | 512                       | 1,2270757                      |
|                            | 1024                      | 0,6831508                      |
| 25                         | 256                       | 1,7569288                      |
|                            | 512                       | 1,1382232                      |
|                            | 1024                      | 0,6685542                      |
| 40                         | 256                       | 1,6771253                      |
|                            | 512                       | 1,1078020                      |
|                            | 1024                      | 0,6641927                      |
| 60                         | 256                       | 1,6091198                      |
|                            | 512                       | 1,0754175                      |
|                            | 1024                      | 0,6632137                      |

#### 4.6.2.2.3 Delay Transmisi

*Delay* transmisi adalah waktu yang diperlukan paket data untuk melalui media transmisi dari satu node ke node yang lain. Dipengaruhi ukuran paket data dan kapasitas kanal intrenet.

- Panjang header IPv4 ( $L_{Header\ Ipv4}$ ) = 20 byte/paket
- Panjang header NALU ( $L_{Header\ NALU}$ ) = 1 byte/paket
- Panjang header UDP ( $L_{Header\ UDP}$ ) = 3 byte/paket
- Panjang header RTP ( $L_{Header\ RTP}$ ) = 12 byte/paket
- Panjang header Ethernet ( $L_{Header\ Ethernet}$ ) = 14 byte/paket
- Panjang header CRC ( $L_{Header\ CRC}$ ) = 4 byte/paket
- Panjang header VLAN ( $L_{Header\ VLAN}$ ) = 4 byte/paket

Perhitungan *delay* enkapsulasi adalah sebagai berikut,

- Jumlah paket yang dikirim (N) = 16.181 paket
- *Average Packet Size* = 1.344,5 byte  
= 10.756 bit
- Jumlah total data yang dikirim = 174.702.696 bit
- Waktu pengiriman total data = 240 s

- Kapasitas Kanal = 100 Mbps  
=  $10^8$  bps

Sehingga, perhitungan *delay* transmisi adalah sebagai berikut:

$$t_t = \frac{(L + L')}{B}$$

$$t_t = \frac{(10.756 + 504)}{10^8}$$

$$t_t = 0,0000148195 \text{ sekon}$$

$$t_t = \mathbf{0,148195 \text{ ms}}$$

Tabel 4.10 berikut merupakan hasil nilai *delay* proses pada masing-masing *Frame rate* dan *bit rate* video.

Tabel 4.10 Nilai *Delay* Transmisi

| <i>Frame Rate</i><br>(Fps) | <i>Bit Rate</i><br>(Kbps) | <i>delay</i><br>Transmisi<br>(ms) |
|----------------------------|---------------------------|-----------------------------------|
| 10                         | 256                       | 0,149097                          |
|                            | 512                       | 0,141258                          |
|                            | 1024                      | 0,131874                          |
| 25                         | 256                       | 0,157377                          |
|                            | 512                       | 0,141029                          |
|                            | 1024                      | 0,136642                          |
| 40                         | 256                       | 0,163971                          |
|                            | 512                       | 0,148195                          |
|                            | 1024                      | 0,139351                          |
| 60                         | 256                       | 0,181487                          |
|                            | 512                       | 0,160190                          |
|                            | 1024                      | 0,158512                          |

#### 4.6.2.2.4 *Delay* Antrian

*Delay* antrian adalah lamanya waktu yang diperlukan paket data untuk diproses pada suatu perangkat, terkait dengan pengiriman dan penerimaan paket data. Lamanya waktu antrian bergantung pada kecepatan saluran dan kondisi antrian. Diketahui dari data primer untuk perhitungan *delay* antrian diantaranya adalah sebagai berikut:

- Jumlah paket yang dikirim (N) = 16.181 paket
- *Average Packet Size* = 1.344,5 byte  
= 10.756 bit

- Jumlah total data yang dikirim = 174.702.696 bit
- Waktu pengiriman total data = 240 s
- Kapasitas Kanal = 100 Mbps  
=  $10^8$  bps

Sehingga, perhitungan kecepatan kedatangan paket pada server ( $\lambda_p$ ) dan kecepatan pelayanan server ( $\mu$ ) adalah sebagai berikut:

$$\lambda_p = \frac{N}{T}$$

$$\lambda_p = \frac{16181}{240}$$

$$\lambda_p = \mathbf{67,42 \text{ paket/sekon}}$$

Sedangkan,

$$\mu = \frac{C}{L_t}$$

$$\mu = \frac{10^8}{(1.344,5 + 63)}$$

$$\mu = \mathbf{9297,14 \text{ paket/sekon}}$$

Perhitungan nilai faktor utilitas( $\rho$ ) adalah sebagai berikut:

$$\rho = \frac{\lambda}{\mu}$$

$$\rho = \frac{67,42}{9297,14}$$

$$\rho = \mathbf{0,007251785}$$

Jadi, perhitungan *delay* antrian adalah sebagai berikut:

$$t_{queue} = \frac{\frac{1}{\mu}}{(1 - \rho)}$$

$$t_{queue} = \frac{\frac{1}{9297,14}}{(1 - 0,007251785)}$$

$$t_{queue} = \mathbf{0,128346 \text{ ms}}$$

Tabel 4.11 berikut merupakan hasil nilai *delay* antrian pada masing-masing *Frame rate* dan *bit rate* video.

Tabel 4.11 Nilai *delay* Antrian

| Frame Rate<br>(Fps) | Bit Rate<br>(Kbps) | <i>delay</i> Antrian<br>(ms) |
|---------------------|--------------------|------------------------------|
| 10                  | 256                | 0,110017                     |
|                     | 512                | 0,108614                     |
|                     | 1024               | 0,098417                     |
| 25                  | 256                | 0,129521                     |
|                     | 512                | 0,118489                     |
|                     | 1024               | 0,106289                     |
| 40                  | 256                | 0,130096                     |
|                     | 512                | 0,128346                     |
|                     | 1024               | 0,117275                     |
| 60                  | 256                | 0,159328                     |
|                     | 512                | 0,137385                     |
|                     | 1024               | 0,125355                     |

#### 4.6.2.2.5 Delay Propagasi

*Delay* propagasi adalah waktu yang dibutuhkan untuk merambatkan paket *multimedia* melalui media transmisi dari *server* ke *client*. Jarak yang dibutuhkan antara pc server dengan *wireless router* adalah 3 meter. Sementara jarak antara *wireless router* dengan *client* 3 meter. Sehingga nilai *delay* propagasi untuk jaringan *wireless local area network* dapat dihitung dengan menggunakan persamaan (2.8).

$$\begin{aligned}
 t_p &= \frac{d_{max}}{v} \\
 &= \frac{6}{3 \times 10^8} = 0,00000002 \text{ s} = 0,00002 \text{ ms}
 \end{aligned}$$

#### 4.6.2.2.6 Delay end-to-end total

*Delay end-to-end* terdiri dari *delay* CODEC dan *delay* jaringan. Dari hasil perhitungan yang telah dilakukan maka dengan persamaan 2-2 besarnya *delay end-to-end* dapat dihitung.

$$\begin{aligned}
 t_{end-to-end} &= t_{codec} + t_{jaringan} \\
 &= 104,6 + (1,107802 + 0,148195 + 0,128346 + 0,00002)
 \end{aligned}$$

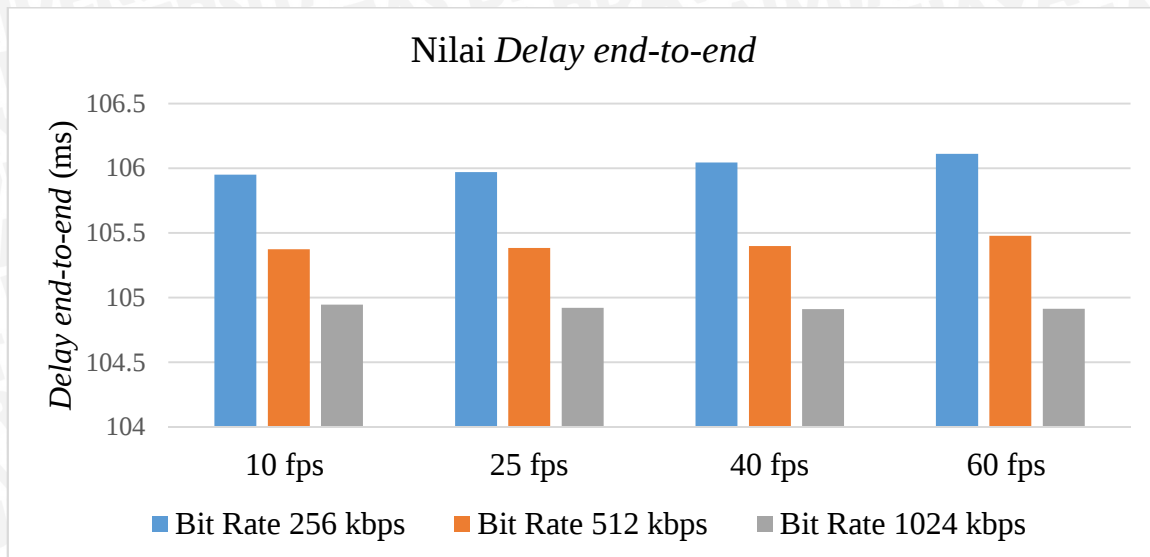
$$= 105,384342 \text{ ms}$$

Tabel 4.12 berikut merupakan hasil nilai *delay end-to-end* pada masing-masing *Frame rate* dan *bit rate* video.

Tabel 4.12 Nilai *Delay end-to-end*

| Frame rate (Fps) | Bit rate (Kbps) | $t_{end-to-end} \{ms\}$ |
|------------------|-----------------|-------------------------|
| 10               | 256             | 105,9499354             |
|                  | 512             | 105,3729931             |
|                  | 1024            | 104,9470808             |
| 25               | 256             | 105,971193              |
|                  | 512             | 105,3843423             |
|                  | 1024            | 104,9208195             |
| 40               | 256             | 106,0438264             |
|                  | 512             | 105,3977416             |
|                  | 1024            | 104,9114854             |
| 60               | 256             | 106,1118244             |
|                  | 512             | 105,4769483             |
|                  | 1024            | 104,9134416             |

Kemudian jika nilai *delay end-to-end* direpresentasikan ke dalam bentuk grafik hubungan antara *Bit rate* dan *Frame rate* ditunjukkan pada Gambar 4.27



Gambar 4.27 Grafik hubungan antara *Bit rate* dan *Frame rate* video dengan *delay end-to-end* secara teoritis

Dari hasil perhitungan diatas, dapat dianalisis bahwa :

1. *Delay end to end* pada WLAN 802.11n dipengaruhi oleh *delay CODEC* dan *delay jaringan* WLAN 802.11n.
2. *Delay jaringan* dalam perhitungan di atas meliputi *delay* proses yang dipengaruhi oleh kecepatan pemrosesan pada pengirim dan penerima, *delay* transmisi yang dipengaruhi oleh kecepatan pemrosesan di setiap node, *delay* propagasi yang dipengaruhi jarak antar node dan media propagasi yang digunakan, serta *delay* antrian yang dipengaruhi oleh faktor utilitas pada jaringan.
3. *Delay end to end* juga dipengaruhi oleh *Frame rate* dan *Bit rate* video semakin tinggi nilai *frame rate* maka semakin besar *delay end-to-end*. Tetapi berbeda halnya dengan semakin tinggi *bit rate* maka nilai *delay end-to-end* semakin kecil.

#### 4.6.2.3 Packet Loss

*Packet loss* dapat dihitung menggunakan persamaan (2.10) :

$$\rho_{total} = 1 - [(1 - \rho_{network})(1 - \rho_{UDP/RTP/IP})]$$

sementara untuk  $\rho_{network}$  dapat dihitung menggunakan persamaan (2.11):

$$P_{network} = \frac{e^{-\frac{Eb}{No}}}{2\sqrt{\pi}\left(\frac{Eb}{No}\right)}$$

Dimana :

$$\frac{Eb}{No} = SNR - 10 \log \frac{B}{R}$$

Untuk SNR WLAN 802.11n adalah 17 dB. Sedangkan R adalah *bit rate* yang di variasikan sebesar 256 Kbps, 512 Kbps, 1024 Kbps. Sedangkan B adalah kapasitas kanal.

$$\frac{E_b}{N_0} = 17 - 10 \log \frac{1.000.000.000}{512.000}$$

$$\frac{E_b}{N_0} = 14,09 \text{ dB}$$

maka  $P_{network}$  adalah :

$$P_{network} = \frac{e^{-14,09}}{2 \sqrt{3,14(14,09)}}$$

$$P_{network} = 5,06369 \times 10^{-4}$$

Probabilitas *packet loss video streaming* dengan *header*UDP/RTP/IP sebesar 60 *byte* (*header* UDP 8 *byte*, *header* RTP 12 *byte*, dan *header* IP 40 *byte*) serta besar *payload video* 254*byte* dan *payload audio* 63 *byte*, yaitu :

$$\begin{aligned} \rho_{UDP/RTP/IP} &= P_{size} \cdot \rho_b \\ &= (60 + 254 + 63) \times 8 \times 10^{-7} \\ &= 3,016 \times 10^{-4} \end{aligned}$$

Sehingga probabilitas *packet loss* pada layanan *live streaming* pada WLAN 802.11n dapat dihitung dengan persamaan.

$$\begin{aligned} \rho_{tot} &= 1 - [(1 - \rho_{net})(1 - \rho_{UDP/RTP/IP})] \\ &= 1 - [(1 - 5,06369 \times 10^{-4})(1 - 3,016 \times 10^{-4})] \\ &= 8,08 \times 10^{-3} \end{aligned}$$

Dan nilai prosentase *packet loss* yang didapat adalah sebagai berikut:

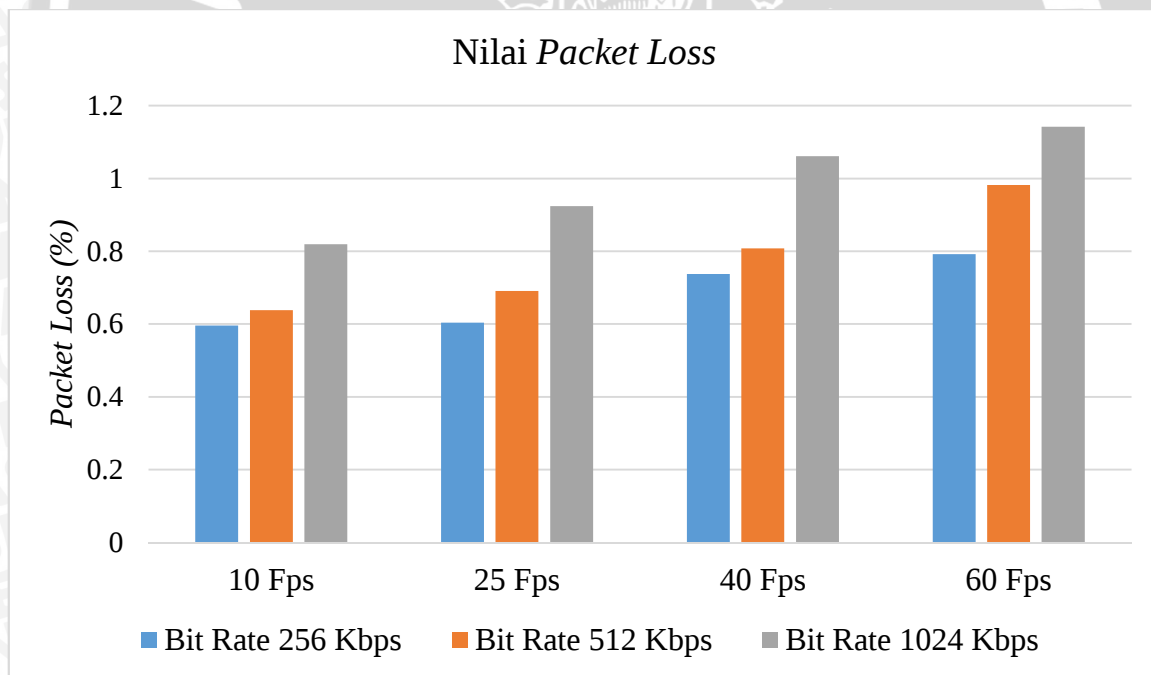
$$\begin{aligned} \rho_{tot}(\%) &= \rho_{tot} \times 100 \% \\ &= 0,000808 \times 100 \% \\ &= 0,808 \% \end{aligned}$$

Dari perhitungan, maka didapatkan nilai *packet loss* pada Tabel 4.13 untuk masing-masing *Frame rate* dan *bit rate* yaitu :

Tabel 4.13 Nilai *Packet loss*

| Frame rate (Fps) | Bit rate (Kbps) | Packet loss (%) |
|------------------|-----------------|-----------------|
| 10               | 256             | 0,596           |
|                  | 512             | 0,638           |
|                  | 1024            | 0,819           |
| 25               | 256             | 0,604           |
|                  | 512             | 0,691           |
|                  | 1024            | 0,924           |
| 40               | 256             | 0,738           |
|                  | 512             | 0,808           |
|                  | 1024            | 1,061           |
| 60               | 256             | 0,792           |
|                  | 512             | 0,982           |
|                  | 1024            | 1,142           |

Kemudian jika nilai *packet loss* direpresentasikan ke dalam bentuk grafik hubungan antara *Bit rate* dan *Frame rate* ditunjukkan pada Gambar 4.28



Gambar 4.28 Grafik hubungan antara *Bit rate* dan *Frame rate* video dengan *Packet loss* secara teoritis

Dari hasil perhitungan diatas, dapat dianalisis bahwa :

1. *Packet loss* pada perhitungan didapatkan nilai dengan pengaruh frame rate dan bit rate yaitu semakin besar *frame rate* dan *bit rate* maka *Packet loss* semakin besar. Hal ini juga dipengaruhi nilai Eb/No.

2. Pada hasil perhitungan *packet loss* didapatkan nilai kategori bagus menurut standar ITU-T-G1010 yaitu dengan standar 0-5 %

#### 4.6.2.4 Throughput

*Throughput* adalah jumlah rata-rata paket yang sukses diterima atau dikirimkan oleh saluran penerima atau pemancar per detik. . Besarnya nilai *throughput* pada *live streaming* dapat ditentukan dengan persamaan (2-10)

$$\lambda = \frac{(1 - \rho)}{t_{trans}[1 + (\alpha - 1)\rho]}$$

Dari nilai *packet loss* diatas, maka nilai *throughput* dapat diketahui. Nilai konstanta propagasi dapat di dapat dari persamaan (2-11).

$$\begin{aligned}\alpha &= \left(1 + \frac{2t_{prop} + 2t_{trans}}{t_{trans}}\right) \\ &= \left(1 + \frac{2 \times 0,00000002 \times 10^{-3} + 2 \times 0,148195 \times 10^{-3}}{0,148195 \times 10^{-3}}\right) = 3,000283629\end{aligned}$$

Sehingga nilai *throughput* dapat diketahui dengan menggunakan persamaan 2.14.

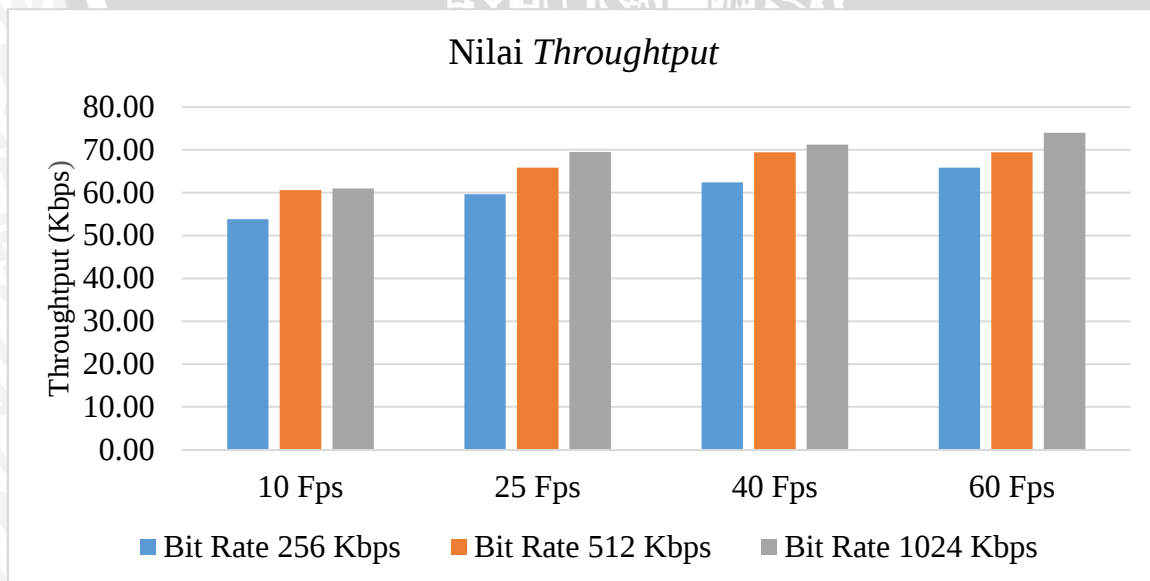
$$\begin{aligned}\lambda &= \frac{(1 - \rho)}{t_{trans}[1 + (\alpha - 1)\rho]} \\ &= \frac{(1 - 0,808)}{0,148195 \times 10^{-3} [1 + (3,000269915 - 1) 0,808]} \\ &= \frac{(0,99404)}{0,000148195 \times [1,351321]} \\ &= \frac{(0,82459)}{(0,0689762 \times 10^{-3})} = 69457,21828 \text{ bps} = 69,46 \text{ Kbps}\end{aligned}$$

Dari perhitungan, maka didapatkan nilai *packet loss* pada Tabel 4.14 untuk masing-masing *Frame rate* dan *bit rate* yaitu :

Tabel 4.11 Nilai *Throughput*

| <i>Frame rate</i><br>(Fps) | <i>Bit rate</i><br>(Kbps) | <i>Throughput</i><br>(Kbps) |
|----------------------------|---------------------------|-----------------------------|
| 10                         | 256                       | 53,81                       |
|                            | 512                       | 60,62                       |
|                            | 1024                      | 60,97                       |
| 25                         | 256                       | 59,66                       |
|                            | 512                       | 65,87                       |
|                            | 1024                      | 69,52                       |
| 40                         | 256                       | 62,40                       |
|                            | 512                       | 69,46                       |
|                            | 1024                      | 71,19                       |
| 60                         | 256                       | 65,89                       |
|                            | 512                       | 69,45                       |
|                            | 1024                      | 74,00                       |

Kemudian jika Tabel 4.13. direpresentasikan ke dalam bentuk grafik hubungan antara besar *frame rate* dan *bit rate* dengan *throughput* secara teoritis ditunjukkan pada Gambar 4.29



Gambar 4.29 Grafik hubungan antara *Bit rate* dan *Frame rate* video dengan *Throughput* secara teoritis

Dari perhitungan *throughput* di atas, dapat dianalisis bahwa :

1. *Throughput* dipengaruhi oleh *delay* transmisi dan *delay* propagasi WLAN 802.16n.
2. Semakin besar nilai *frame rate* dan *bit rate* maka nilai *throughput* semakin besar, hal ini mempengaruhi kecepatan WLAN 802.11 n dalam pengiriman data.

#### 4.7. Analisis dan Pembahasan Data

Dari hasil simulasi dan perhitungan didapatkan perbandingan nilai yaitu :

1. Pada analisis simulasi *delay* menggunakan *packet analyzer* Wireshark didapatkan nilai *delay* yang relatif kecil dengan video dengan *frame rate* 10 didapatkan nilai <150 ms untuk semua variasi *bit rate* video . Untuk *frame rate* 25 didapatkan nilai <150 ms untuk semua variasi *bit rate* video. Untuk *frame rate* 40 didapatkan nilai <150 ms untuk semua variasi *bit rate* video. Untuk *frame rate* 60 didapatkan nilai <150 ms untuk semua variasi *bit rate* video. Dimana ketika nilai *frame rate* semakin besar maka *delay* yang dihasilkan dari *capturing* wireshark semakin besar. Sementara ketika nilai *bit rate* semakin tinggi nilai *delay* yang dihasilkan dari *capturing* wireshark semakin kecil. Sedangkan untuk hasil perhitungan *delay end-to-end* didapatkan nilai *delay* <150 ms untuk variasi *frame rate* dan *bit rate* yang lainnya. Hal ini dipengaruhi oleh kecepatan pemrosesan pada pengirim dan penerima, *delay* transmisi yang dipengaruhi oleh kecepatan pemrosesan di setiap node, sedangkan untuk *delay* propagasi faktor yang mempengaruhi adalah jarak antar node dan media propagasi yang digunakan. Sehingga dari perbandingan tersebut nilai untuk data berdasarkan *capturing* wireshark dan perhitungan tergolong kategori bagus menurut standar ITU-T G.114 karena nilai *delay* berkisar 0-150 ms.
2. Pada analisis simulasi *Packet loss* menggunakan *packet analyzer* Wireshark didapatkan nilai untuk video dengan *frame rate* 10, 25, 40 dan semua variasi *bit rate* nilai *packet loss* baik karena memiliki nilai <5%. Sementara untuk nilai *frame rate* 60 dan *bit rate* 256 Kbps, 512 Kbps memiliki nilai *packet loss* yang baik karena nilai *packet loss* < 5%, tetapi untuk *bit rate* 1024 Kbps nilai *packet loss* 5.77% berdasarkan ITU-T-1010 *packet loss* dikategorikan cukup. Berdasarkan data hasil *capturing* wireshark dapat disimpulkan bahwa semakin besar nilai *frame rate* dan *bit rate* video nilai *packet loss* semakin besar. Hal ini disebabkan oleh kemungkinan jumlah paket yang besar akan kehilangan paket yang besar juga. Sedangkan berdasarkan hasil perhitungan nilai *packet loss* untuk semua *frame rate* dan variasi *bit rate* menunjukkan

nilai  $<5\%$  maka baik berdasarkan ITU-T-100. Nilai ini dipengaruhi oleh nilai SNR dan Eb/No.

3. Pada analisis simulasi *throughput* menggunakan *packet analyzer* Wireshark dan perhitungan didapatkan nilai dimana semakin besar nilai *frame rate* dan *bit rate* video maka *throughput* semakin besar. Hal ini disebabkan jumlah *frame* dan *bit rate* video. Jumlah *frame* dan *bit rate* pada video yang besar menunjukkan nilai yang besar pula, hal ini yang mengakibatkan nilai *throughput* semakin besar. Sedangkan untuk perhitungan faktor yang mempengaruhi antara lain *delay* propagasi dan *delay* transmisi. Sehingga *throughput* yang merupakan banyak bit yang dapat ditransmisikan dan sukses diterima di tujuan per detik maka jika *throughput* semakin besar kecepatan pengiriman dan penerimaan data semakin cepat.



## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan meliputi pengamatan terhadap performansi layanan *live streaming* pada jaringan WLAN 802.11n, dapat disimpulkan bahwa,

1. Konfigurasi sistem *live streaming* pada jaringan *Wireless Local Area Network* (WLAN) telah berhasil dibangun. Hal ini dapat dibuktikan dari hasil pengujian untuk koneksi maupun berjalannya video pada *client*.
2. Kualitas performansi layanan *live streaming* berdasarkan parameter *packet loss*, *delay end-to-end*, dan *throughput* memiliki perbedaan nilai untuk hasil pengamatan dan hasil perhitungan. Tetapi hal ini tidak menurunkan kualitas layanan *live streaming* pada jaringan WLAN 802.11n
3. Kualitas performansi layanan *live streaming* pada jaringan *Wireless Local Area Network* 802.11n adalah sesuai dengan standar ITU-T G.114 untuk *delay* (*delay end to end* < 150 s) pada setiap variasi *frame rate* dan *bit rate* video. Dan probabilitas *packet loss* (*packet loss* 5-10%) untuk video dengan *frame rate* 60 dan *bit rate* 1024 Kbps Kualitas layanan cukup berdasarkan ITU-T G.1010, Sementara untuk variasi *frame rate* dan *bit rate* lainnya nilai *packet loss* (*packet loss* <5%) maka kualitas layanan berjalan baik. Berdasarkan hasil perhitungan kualitas performansi layanan *live streaming* berjalan dengan baik dimana *delay end-to-end* setiap variasi *frame rate* dan *bit rate* memiliki nilai (<150 ms). Sementara nilai *packet loss* (<5%) maka kualitas layanan *live streaming* pada jaringan WLAN 802.11n berjalan dengan baik.

#### 5.2 Saran

Saran yang dapat diberikan untuk pengembangan skripsi ini selanjutnya adalah :

1. Jumlah *client* pada jaringan WLAN ditambah agar dapat mengoptimalkan sesuai dengan standar 802.11n
2. Menggunakan CODEC dan jenis jaringan yang berbeda.
3. Dalam penelitian ini *client* tidak bergerak. Baiknya untuk penelitian selanjutnya *client* bergerak.

## DAFTAR PUSTAKA

- Austerberry, David. 2004. *The Technology of Video & Audio Streaming (2nd Edition)*. London: Jordan Hill.
- Apostolopoulos, John G. 2001. *Video Communication and Video Streaming*. Hewlett-Packard Laboratories: Streaming Media System Group
- Forouzan A Behrouz. 2007. *Data Communication and Networking Fourth Edition*. New York: McGraw-Hill Companies, Inc.
- IEEE 802.11 ([https://en.wikipedia.org/wiki/IEEE\\_802.11](https://en.wikipedia.org/wiki/IEEE_802.11)) diakses pada 3 april 2016
- IEEE 802.11n-2009—Amendment 5: Enhancements for Higher Throughput". IEEE-SA. 29 October 2009.
- ITU-T. 2003. Recommendation ITU-T G.114: One-Way Transmission Time. (Online). (<http://www.itu.int/rec/T-REC-G.114-200305-I/en>, diakses tanggal 1 april 2016)
- ITU-T Study Group 12. 2001. Recommendation G.1010 - End User Multimedia QoS Categories. (Online). (<http://www.itu.int/rec/T-REC-G.1010-200111-I/en>, diakses tanggal 1 april 2016).
- Salkintzis A, Passas N. 2005. *Emerging Wireless Multimedia Services and Technologies*. Chichester: John Willey & Son.
- Schwartz, Mischa. 2005. *Mobile wireless communication*. New York: Cambridge University Press.
- Stalling, William. 1997. *Data & Computer Communications Fifth Edition*. New Jersey: Prentice-Hall, Inc.
- Understanding the IEEE 802.11n Amendment (Understanding the IEEE 802.11n Amendment, diakses tanggal 14 agustus 2016)

## LAMPIRAN 1

### SPESIFIKASI WIRELESS ROUTER

| Spesifikasi                  | Keterangan                                                                                                   |
|------------------------------|--------------------------------------------------------------------------------------------------------------|
| <i>Standards</i>             | IEEE 802.11n, IEEE 802.11g, IEEE 802.11b                                                                     |
| <i>Interface</i>             | 4 10/100M Auto-Sensing RJ45 LAN Port(Auto MDI/MDIX)<br>1 10/100M Auto-Sensing RJ45 WAN Port(Auto MDI/MDIX)   |
| <i>Wireless Signal Rates</i> | Up to 150Mbps                                                                                                |
| <i>Frequency Range</i>       | 2.4-2.4835GHz                                                                                                |
| <i>EIRP</i>                  | <20dBm(EIRP)                                                                                                 |
| <i>Wireless Functions</i>    | Enable/Disable Wireless Radio, WDS Bridge, WMM, Wireless Statistics                                          |
| <i>Receiver Sensitivity</i>  | 130M: -68dBm@10% PER<br>54M: -68dBm@10% PER<br>11M: -85dBm@8% PER<br>6M: -88dBm@10% PER<br>1M: -90dBm@8% PER |
| <i>Antenna Type</i>          | 5dBi Detachable Omni Directional Antenna                                                                     |
| <i>Software Functions</i>    | NAT, DoS Firewall, DHCP                                                                                      |
| <i>Operating temperature</i> | 0°C~40°C (32°F~104°F)                                                                                        |
| <i>Protocols</i>             | Supports IPv4 and IPv6                                                                                       |
| <i>Dimensi</i>               | 6.9 x 4.6 x 1.3 in. (174 x 118 x 33 mm)                                                                      |
| <i>Jarak Jangkauan</i>       | 22 meter (Indoor) 50 meter (Outdoor)                                                                         |

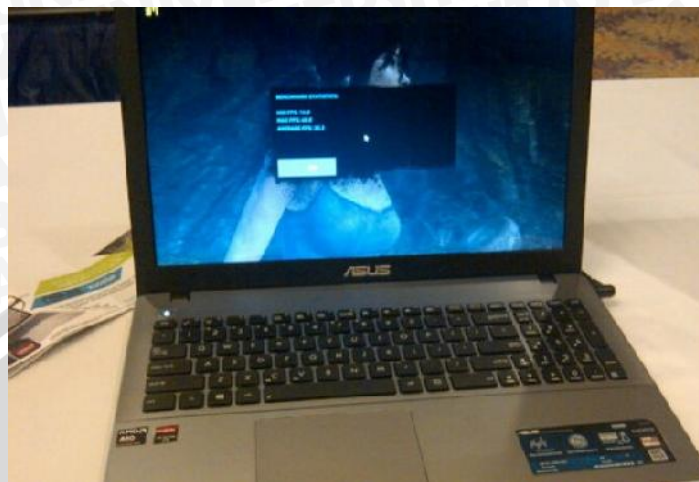
Sumber: <http://www.tp-link.com/>, 2016



Gambar Wireless Router TP-LINK TL-WR74IND

## LAMPIRAN 2

## PC Server



Gambar PC Server ASUS X550

| Spesifikasi    | Keterangan                                              |
|----------------|---------------------------------------------------------|
| Layar          | 15.6 Inch Resolusi HD 1366×768 Rasio 16:9               |
| Chipset        | AMD A76M FCH                                            |
| Processor      | AMD APU A10-5750M                                       |
| VGA            | AMD Radeon HD 8650G + AMD Radeon HD 8670M (Dual Grafis) |
| Penyimpanan    | 500HDD GB                                               |
| Sistem Operasi | Windows 10                                              |

### LAMPIRAN 3

#### PC Client

##### 1. ASUS A46CB



Gambar PC Client 1 ASUS A46CB

| Spesifikasi    | Keterangan               |
|----------------|--------------------------|
| Layar          | 14.0" 16:9 LED Backlight |
| Dimensi        | 24.8 x 2.10 x 34.8 cm    |
| Processor      | Intel Core i3 3217U      |
| VGA            | NVIDIA GeForce GT 635M   |
| Penyimpanan    | 500HDD GB                |
| Sistem Operasi | Windows 8                |

##### 2. HP 4B11AU



Gambar PC Client 2 HP 4B11AU

| Spesifikasi    | Keterangan                             |
|----------------|----------------------------------------|
| Layar          | 14.0" 16:9 Active Matrix TFT Color LCD |
| Dimensi        | 24.8 x 2.10 x 34.8 cm                  |
| Processor      | Intel Core i3 3217U                    |
| VGA            | NVIDIA GeForce GT 610M                 |
| Penyimpanan    | 500HDD GB                              |
| Sistem Operasi | Windows 7 64 bit                       |

### 3. LENOVO YP410



Gambar PC Client 3 LENOVO YP410

| Spesifikasi    | Keterangan                                                     |
|----------------|----------------------------------------------------------------|
| Layar          | 14.0" HD+ Glossy LED Backlit with integrated camera (1600x900) |
| Dimensi        | 13.77 x 9.64 x 0.59-1.29 inches                                |
| Processor      | Intel Core i7-4700MQ                                           |
| VGA            | NVIDIA GeForce GT 755M 2GB                                     |
| Penyimpanan    | 1 HDD TB                                                       |
| Sistem Operasi | Windows 8.1 64 bit                                             |

#### 4. ASUS KD5D



Gambar PC Client 4 ASUS KD5D

| Spesifikasi    | Keterangan             |
|----------------|------------------------|
| Layar          | 14.0" Full HD          |
| Dimensi        | 382 x 255 x 21.7 cm    |
| Processor      | Intel Core i5-5200U    |
| VGA            | NVIDIA GeForce GT 725M |
| Penyimpanan    | 1 HDD TB               |
| Sistem Operasi | Windows 8.1 64 bit     |

#### 5. TOSHIBA SATELITE L745



Gambar PC Client 5 TOSHIBA SATELITE L745

| Spesifikasi    | Keterangan             |
|----------------|------------------------|
| Layar          | 14" WXGA LED           |
| Dimensi        | 382 x 255 x 21.7 cm    |
| Processor      | Intel Core i5-2410M    |
| VGA            | NVIDIA GeForce GT 525M |
| Penyimpanan    | 640 HDD GB             |
| Sistem Operasi | Windows 7 64 bit       |



## Lampiran 4

Pengaruh jumlah client dengan performansi jaringan WLAN 802.11n

| <i>Client</i> | <i>Delay end-t- end (ms)</i> | <i>Packet loss (%)</i> |
|---------------|------------------------------|------------------------|
| 1             | 83.05342                     | 0.42016                |
| 2             | 88.63950                     | 0.51712                |
| 3             | 94.22558                     | 0.61408                |
| 4             | 99.81166                     | 0.71104                |
| 5             | 105.39774                    | 0.80821                |
| 6             | 110.98382                    | 1.11504                |
| 7             | 116.56990                    | 2.06848                |
| 8             | 122.15598                    | 2.69872                |
| 9             | 127.74206                    | 3.32896                |
| 10            | 133.32814                    | 3.9592                 |
| 11            | 138.91422                    | 4.58944                |
| 12            | 144.50030                    | 5.21968                |
| 13            | 150.08638                    | 5.84992                |
| 14            | 155.67246                    | 6.48016                |
| 15            | 161.25854                    | 7.1104                 |
| 16            | 166.84462                    | 7.74064                |
| 17            | 172.43071                    | 8.37088                |
| 18            | 178.01679                    | 9.00112                |
| 19            | 183.60287                    | 9.63136                |
| 20            | 189.18895                    | 10.2616                |



## Lampiran 5

Pengaruh jarak dengan performansi jaringan WLAN 802.11n

| Jarak (m) | <i>Delay end-to-end</i> (ms) | <i>Packet loss</i> (%) |
|-----------|------------------------------|------------------------|
| 1         | 96.96592                     | 0.68680                |
| 2         | 101.18183                    | 0.74740                |
| 3         | 105.39774                    | 0.80821                |
| 4         | 109.61365                    | 1.44333                |
| 5         | 113.82956                    | 2.07866                |
| 6         | 118.04547                    | 2.71399                |
| 7         | 122.26138                    | 3.34932                |
| 8         | 126.47729                    | 3.98465                |
| 9         | 130.69320                    | 4.61998                |
| 10        | 134.90911                    | 5.25531                |
| 11        | 139.12502                    | 5.89064                |
| 12        | 143.34093                    | 6.52597                |
| 13        | 147.55684                    | 7.16130                |
| 14        | 151.77275                    | 7.79663                |
| 15        | 155.98866                    | 8.43196                |
| 16        | 160.20457                    | 9.06730                |
| 17        | 164.42048                    | 9.70263                |
| 18        | 168.63639                    | 10.33796               |
| 19        | 172.85230                    | 10.97329               |
| 20        | 177.06821                    | 11.60862               |
| 21        | 181.28412                    | 12.24395               |
| 22        | 185.50003                    | 12.87928               |